

Nr. 5/86 Mai



IM TEST : DIE SENSATION 128 K-GRAM KARTE

IM TEST: TI-Artist

SERVICE: 4. Laufwerk kein Problem

IM TEST: Monitor

SERVICE: Assembler

IM TEST: Multiplan

SERVICE: Listschutz für Basic-Programme

IMPRESSUM

TI-REVUE, die Zeitschrift für den TI PC und TI 99/4A erscheint monatlich in der AKTUELL-GRUPPE, Elisabethstraße 1, 8044 Lohhof.
Redaktion: Senator-Presse-Service.
Verantwortlich für den Inhalt: Heiner Martin.
Verantwortlich für Anzeigen: Bruno Redase.
Verantwortlich für Listings: Harmut König.
Alle: Postfach 1107, 8044 Lohhof. Anfragen bitte nur schriftlich.

Druck: Maier und Söhne
Es gilt die Honorarliste des Verlages. Für unaufgefordert eingesandte Manuskripte und Listings keine Haftung.
Bei Einsendung von Texten, Fotos und Programmträgern erteilt der Autor dem Verlag die Genehmigung für einen einmaligen Abdruck sowie die Aufnahme in den Programm-Service nach den Verlags-Sätzen!

Namentlich gezeichnete Artikel geben nicht unbedingt die Meinung der Redaktion wieder.
Kein Anspruch auf Lieferung bei Ausfall durch höhere Gewalt.
Gerichtsstand: München
Geschäftsführer: Werner E. Seibt
Abo- und Kassetten-Service: Henny Rose Seibt

INHALT

Nr. 5/86 Mai

DIALOG

TEST & TECHNIK

SERVICE

LISTINGS

Grüß Gott Guten Tag

Die Gerüchteküche brodelt ja mal wieder kräftig. Um es gleich vorweg zu nehmen, und um auch für mehr Klarheit zu sorgen, auch wenn es uns TI-User nicht so arg glücklich macht: Die Fa. Myarc hat auch auf der TI-Ausstellung Anfang März in Los Angeles den Nachfolgecomputer nicht gezeigt.
Weiterhin ist bis heute das schon lange angekündigte, auch seit mehreren Monaten in den Preislisten amerikanischer Händler zu findende Extended Basic Level IV (ebenfalls von Myarc entwickelt) bis heute nicht ausgeliefert. So wollen wir in der TI-REVUE weiterhin nur Produkte testen, die auf dem Markt verfügbar sind und bei denen unser Testmodell elektrisch wie in der Software exakt dem ausgelieferten Produkt entspricht.
Entwicklungsmustern oder Gerüchten hinterher zu jagen nützt uns allen nichts, das steigert höchstens die Verunsicherung.
Einem weiteren Gerücht müssen wir hier leider entgegentreten, aufgrund schon diverser Anfragen durch unser Lesertelefon: Die neuen MSX-2-Computer benutzen einen Video-Display-Prozessor mit der Bezeichnung V9938. Dieser sei angeblich 100% kompatibel zum TMS 9928 des TI 99/4A. Nun, unter vollständiger Kompatibilität versteht man ja wohl Herausnehmen des alten und Einsetzen des neuen IC's.
Das geht beim 9938 auf gar keinen Fall, denn er besitzt 64 Anschlüsse, im Gegensatz zu den 40 des 9928. Auch hinsichtlich der Software bestehen dabei durchaus Unterschiede. Das muß nicht bedeuten, daß dieser Chip nicht durchaus für den TI 99/4A geeignet ist, aber es bedarf in jedem Fall zusätzlicher Hard- und Software, die erst einmal entwickelt werden muß.
Wenden wir uns aber „wichtigeren" Dingen zu. Manchmal kommt auch einer Zeitschrift der Zufall zu Hilfe. Von der schon erwähnten TI-Ausstellung in Los Angeles haben wir uns ein neues Grafikprogramm mitbringen lassen.
Der Zufall wollte es, daß wir eine gerade erst fertiggestellte neue Version des TI-Artist, wie sich das Programm nennt, erhielten. Den Testbericht lesen Sie natürlich in dieser Ausgabe und bis zum nächsten Monat wünschen wir Ihnen viel Spaß mit dem TI 99/4A.
Ihr TI-REVUE-Team

Nicht vergessen: Seid Ihr mit dem Heft zufrieden, sagt es weiter, seid Ihr unzufrieden, sagt es uns. Und: Jeden Dienstag von 15 bis 19 Uhr stehen Ihnen unsere Lesertelefone zur Verfügung. Für technische und Assembler-Fragen Tel.-Nr. 0731/33220 und zu den Listings bzw. Fragen zu den Abonnements und dem Kassettenservice Tel.-Nr. 089/1298013.

PS: Der Abo- und Versand-Service muß sich wieder mal zu Wort melden. Zunächst mit einer Entschuldigung: Etwa 20 Abonnenten erhielten in den letzten Märztagen Mahnungen, obwohl die Abogebühr bezahlt war. Der Grund: Ein Programmabsturz vernichtete eine ganze Datei auf unserer Festplatte. Und wie das Leben so spielt, hatte der Abo-Service auch zwei Wochen lang kein Backup gemacht, mußte also auf zwei Wochen alte Daten zurückgreifen. Sorry, daß dabei auch einige Lesernamen mit "Nicht bezahlt" eingegeben wurden, obwohl bezahlt war.
Womit in diesem Zusammenhang erneut die Bitte ausgesprochen wird: Einzahlungsbelege deutlich beschriften bitte, denn mit einer hingekritzelten Linie kann niemand was anfangen! Sie ersparen sich und uns Ärger und Kosten. Wenn's geht, per Schreibmaschine, wenigstens aber in Blockbuchstaben. Bitte.

BRIEFE

UNKONTROLLIERTE ABSTÜRZE

Nachdem bereits einiges über Schwierigkeiten mit der Verbindung Modul (z.B. X-Basic) – TI 99/4A geschrieben wurde, möchte ich auch kurz von meinen ähnlichen, aber nicht identischen Erfahrungen berichten. Plötzliche Abstürze verbunden mit der Darstellung interessanter geometrischer Muster auf dem Bildschirm ließen mich schon recht bald an dem Vorhandensein einer ordnungsmäßigen Verbindung zwischen Modul und Modulportstecker zweifeln. Ich versuchte es mit Kontaktspray, ohne Wirkung. Eine Demontage des Gerätes zeigte, daß sich ein Schaumgummiring, der zum Schutz des Modulschachtes vor Verunreinigungen vor dem Stecker angebracht ist, in Krümel aufgelöst und zwischen die Kontaktflächen geschoben hatte. Dies kann durchaus auch als Folge des Einsatzes von Kontaktspray geschehen sein. Nach restloser Entfernung des Schaumgummiringes und natürlich auch der Krümel im Stecker waren auch die Störungen im Modul-Betrieb des Rechners passe. Nach ca. einem dreiviertel Jahr traten wieder Störungen auf, die sich jedoch problemlos mit Kontaktspray und anschließender Reinigung der Kontaktflächen beseitigen ließen.
Axel Rincke, München

Wir danken für den Hinweis. Die größte Anzahl der Störungen beim TI 99/4 A wird durch Kontaktschwierigkeiten am Modulstecker verursacht. Das kann, insbesondere wenn wie hier geschildert der Schaumstoffring zerfallen ist, sogar soweit führen, daß der Rechner ohne eingestecktes Modul ebenfalls hier und da abstürzt. Deshalb ist eine Reinigung des Modulsteckers und der Kontakte der Module von Zeit zu Zeit notwendig. Wer es genau machen will, sollte auch die Steckverbindung zwischen der Platine, auf der der eigentliche Modulstecker sitzt, und der Zentralplatine des Rechners säubern. Wir möchten hier aber nochmal darauf hinweisen, daß jegliche Mittel, die Öl enthalten, überhaupt nicht geeignet sind. Das, was üblicherweise als Kontaktspray verkauft wird, enthält meist ein sehr leichtes Öl und ist damit für die Bearbeitung des Modulports nicht geeignet. Deshalb sollte hier unbedingt ein ölfreies Reinigungsspray verlangt und eingesetzt werden.

SPEICHERERWEITERUNG DEFEKT?

Wenn bei angeschlossener Speichererweiterung der TI eine DIS/VAR80-Datei mit CALL LOAD ("DSKX.NAME") nicht erkennt und mit I/O ERROR 02 reagiert, alle Dateinamen aber korrekt sind, CALL INIT durchgeführt wurde usw., ist da die Speichererweiterung defekt?
Walter, Dockal, Mödling

Um es vorweg zu nehmen, die Speichererweiterung ist nicht defekt. Mit CALL LOAD lädt der TI Assemblerprogramme, die im "Tagged Object Code" vorliegen. Dieser verwendet die Dateiform DIS/FIX80. Eine DIS/VAR80 Datei kann deshalb kein assembliertes Maschinenprogramm sein, höchstens der dazugehörige Quellcode, aus dem der Assembler dann den "Tagged Object Code" formt. So kommt es bei dem Versuch, mit CALL LOAD eine DIS/VAR Datei zu laden, verständlicherweise zu einer Fehlermeldung. Es erreichen uns auch ab und zu Fragen, warum der TI Assemblerprogramme so langsam lädt. Nun, der Tagged Object Code ist vornehmlich dazu gedacht, die Programme relokatibel zu laden. Was heißt das nun schon wieder? Bei den meisten anderen Computern muß ich für ein Assemblerprogramm einen ganz bestimmten Speicherbereich reservieren und das Programm arbeitet dann auch nur in diesem Bereich. Will ich dann noch ein weiteres Assemblerprogramm laden, welches den gleichen Speicherbereich beansprucht, bekomme ich Ärger. Der TI ist da viel vornehmer: Die Programme werden im Normalfall so assembliert, daß beim Laden zuerst einmal ein freier Speicherplatz gesucht wird und dann alle Sprünge, Labels usw. entsprechend auf den Speicherbereich umgerechnet werden. Das braucht halt mehr Zeit, ich brauche mir aber keine großen Gedanken zu machen, welche Assemblerprogramme ich nun zusammen verwenden will.

FORTH MIT KASSETTE

Ich möchte gerne mit FORTH arbeiten. Da ich keine Diskettenstation besitze, brauche ich die Modulversion. Bei welchem Geschäft kann ich mir diese kaufen? Und was für Geräte brauche ich sonst noch (RAM-Erweiterung, Drucker etc.)? Desgleichen bei LOGO: Geht es auch ohne Floppy? Weiter möchte ich fragen, ob es die Software für normale Floppys auch für das Quick-Disk-System gibt?
Christian Schäfer, Wanfried

Die Modulversion des FORTH wird von der Fa. Schuh, Goethestr. 17, 8034 Germering, vertrieben. Eine RAM-Erweiterung ist für FORTH unbedingt erforderlich, desgleichen für LOGO. Einen Drucker brauchen Sie nicht, er ist aber wohl doch für einen ernsthaften Programmierer schon allein wegen der Listings nützlich. Noch ein Wort zum Betrieb von FORTH mit der Kassette. Da bei FORTH das Programm ja nicht in einem Stück geladen wird, sondern in einzelnen Abschnitten (Screens), ist die ganze Handhabung ziemlich umständlich. Die Quick-Disk Floppy ist bisher nicht ausgeliefert worden. Hier sei uns die Anmerkung gestattet, daß die Redaktion für Anzeigen von Firmen nicht verantwortlich zeichnet. Ob die vorhandene Software dann auch für die Quick-Disk erhältlich ist, wird sich wohl erst zeigen, wenn diese auf dem Markt zu haben ist (falls es überhaupt kommt), und dann werden wir natürlich schnellstens in der TI-REVUE einen Testbericht veröffentlichen.

LANGE VERBINDUNGS–LEITUNGEN

Ich habe drei Probleme:
1. Ich habe das Sonderheft TI-SPEZIAL 2/85 nirgends kaufen können. Können Sie mir sagen, wo ich dieses Sonderheft bekommen könnte?
2. Ich würde gern einen Akustikkoppler an meinen Computer anschließen. Leider beträgt die Entfernung zwischen Konsole und Telefon rund 6 Meter. Kann ich diese Entfernung überbrücken?
3. Gibt der TI 99/4A eine Fehlermeldung "I/O ERROR" aus, so folgt der Meldung eine zweistellige Nummer. Gibt es eine Möglichkeit, daß der TI 99/4A bei Verwendung des CALL ERR Befehls auch diese zweistellige Zahl ausgibt?
Mattias Sorba, Frankfurt

Jetzt freut sich wieder unser Vertrieb: Wenn Sie irgendwelche Sonderhefte oder vergangene Ausgaben nicht mehr beim Zeitschriftenhandel erhalten

können, so wenden Sie sich direkt an unseren Verlag: TI-REVUE, Postfach 1107, 8044 Lohhof. Dort erhalten Sie es bestimmt. Die Verbindungsleitungen zwischen RS232-Schnittstellen, und über diese wird ja ein Akustikkoppler betrieben, können bis max. 20 m lang sein, wobei wir aber empfehlen, bei ca. 10 m Schluß zu machen. 6 m sind aber mit Sicherheit kein Problem. Wenn es dennoch Fehler gibt, sollten abgeschirmte Kabel verwendet werden. Bei Centronics-Anschlüssen ist die max. Länge übrigens auf ca. 2 m begrenzt. Leider gibt es unseres Wissens keine Möglichkeit, bei CALL ERR die zweistellige Kennziffer hinter der I/O ERROR-Meldung zu erhalten. Übrigens ist dies auch bei viel größeren Computern wohl ein Problem, denn hier finden sich in vielen Anwenderprogrammen die lapidare Meldung "Datei kann nicht angesprochen werden". Dieser "Standard" kann beim TI 99/4A mit CALL ERR leicht verwirklicht werden. Nun gibt es da ja noch den Sonderfall, daß die Datei zwar korrekt eröffnet und auch gelesen werden kann, jedoch versucht das Programm, hinter das Dateiende zu lesen. Dies kann aber auch einfach mittels der EOF-Funktion abgefangen werden.

RESET-TASTE

Ich habe die Reset-Taste am Modulport angeschlossen (zwischen Pin 1 und 29). Wenn ich die Taste im TI-Basic drücke (ohne eingestecktes Modul), dann schaltete der Computer auf das Titelbild und das Programm ist gelöscht. Ist aber ein Modul eingeschoben, dann funktioniert diese Taste nicht mehr. Ist das normal? Wird beim Betätigen der Taste immer das Programm gelöscht und auf das Titelbild zurückgeschaltet? Wenn das so ist, wo bleibt da der Sinn dieser Taste, genausogut könnte ich ja einfach ausschalten. Bei anderen Computern bleibt das Programm aber nach Drükken einer solchen Taste erhalten, ist das beim TI auch möglich? Welche Funktion hat die RESET-Leitung am I/O-Port? Gibt es eine Möglichkeit, den TI in seiner Verarbeitungsgeschwindigkeit zu erhöhen? Es ist zwar ein 16-Bitter mit einer Taktfrequenz von 3 MHz, ist aber dennoch langsamer als der C64?
Gerald Fingerlos,
Elsbethen

Da haben wir bei der kurzen Antwort hinsichtlich des Reset-Taster doch etwas vergessen: Da in jedem Modul ein eigener "Reset-Schalter" (einfache Verbindung über 100 Ohm von Pin 1 und 29) eingebaut ist, muß diese, soll der extra angeschlossene Schalter auch bei eingestecktem Modul funktionieren, bei allen Modulen unterbrochen werden. Der Sinn eines Reset-Tasters ist vor allem für Maschinenprogrammierer gegeben. Hier wird dadurch ganz bestimmt der Ein/Aus-Schalter der Konsole geschont, auch bleiben beim Mini-Memory und beim Extended Basic ja auch die Maschinenprogramme erhalten. Bei Basic-Programmen ist das nicht möglich, außer bei Extended Basic mit Speichererweiterung, hier ist eine Hilfsroutine in Maschinensprache denkbar. Die RESET-Leitung am I/O-Port ist quasi der Ausgang des Reset-Tasters, d.h. durch diesen Taster wird ein Hardware-Reset über den Taktgenerator des TMS 9900 durchgeführt. Ein Ausgang des Taktgenerators erzeugt dann einen Reset-Impuls bestimmter Länge, der an den I/O-Port weitergegeben wird, um den angeschlossenen Peripheriegeräten ein Rückstellen auf die Ausgangsparameter zu ermöglichen. Die Geschwindigkeit des TI 99/4A läßt sich mit einfachen Mitteln nicht erhöhen. Das liegt zum einen an den sehr langsamen GROM's, die im TI und auch den Modulen Verwendung finden und zum anderen ist die Zugriffszeit auf dem I/O-Port bzw. auf die Peripheriegeräte nochmal gebremst. Über derartige Sachen redet kein Hersteller gerne, aber die Taktfrequenz eines Computers ist nicht alles. Da gibt es gerade erst wieder ein neues Beispiel, wo ein 6 MHz getakteter Prozessor mit 1 MHz Speicherzyklen fährt. Aber, um es hier nochmal klar und deutlich zu sagen: In Assembler kommt ein C64 nicht mit dem TI 99/4A mit!

ASSEMBLER-POKELISTEN

Ich bin begeisterter TI-REVUE-Leser und beglückwünsche Sie als solcher zu Ihrer hervorragenden Zeitschrift. Nur eine Tatsache ist mir negativ aufgefallen: Die Assembler-Programme kommen in Ihrer Zeitschrift zwar nicht zu kurz, aber sie kommen in der falschen Form. Mit "falscher Form" meine ich die Poke-Listen, denn man muß Pokelisten ja als "Data-Wüste" abdrucken, aus der man nicht schlau wird, und in die man meistens beim Abtippen Fehler einbaut. Wäre es da nicht besser, den Assembler-Quellcode abzudrucken(vielleicht verkleinert)? Oder wenigstens die Poke-Liste mit Prüfsummen abzudrukken, die beim Poken kontrolliert werden?
Peter Kliem,
Erftstadt

In den letzten Ausgaben haben wir verstärkt auf Assembler-Quellcode umgestellt. Dabei gab es eine kleine Hemmschwelle bei unserer Herstellung zu überwinden, die sich mit Basic-Listings nun schon gut „abgefunden" hat. Aber mit den Assembler-Listings sah man hier wohl die Gefahr von zusätzlichen Fehlern. Nun, auch das haben wir (hoffentlich?!) in den Griff bekommen und in Zukunft werden Sie, von Ausnahmefällen abgesehen, den Quellcode in der TI-REVUE finden.

WO BLEIBEN DIE DATEN BEI ERROR DETECTED IN DATA?

Auf unserem Anrufbeantworter der technischen Redaktion fanden wir folgende Frage, leider ohne Namen und Anschrift des Fragestellers. Dabei gleich eine Bitte: Wer hier seine Fragen aufspricht, erhält von uns natürlich eine schriftliche Antwort, aber eben nur, wenn er die komplette Anschrift nennt.

Gibt es eine Möglichkeit, wenn beim Laden eines Programms von Kassette mittendrin die Meldung ERROR DETECTED IN DATA erscheint, die bis dahin gelesenen Daten sichtbar zu machen, bzw. zu retten, damit dann bei einem Basicprogramm nur ein paar Zeilen nachgetragen werden müssen?

Beim TI werden Basic-Programme nicht der Reihenfolge der Zeilen nach im Speicher abgelegt, sondern es kommt zuerst eine Zeilenliste, die jeweils aus Zeilennummer und einem Pointer besteht, der auf den Inhalt der Zeile im Speicher zeigt. Darüber befinden sich dann die Inhalte der Zeilen, aber nicht in geordneter Form, sondern in der Reihenfolge der Eingabe bzw. des Veränderns der Zeilen. Beim Absaven wird aber der ganze vom Basic-Programm belegte Speicherplatz in einem Rutsch auf Kassette gespeichert.

Wenn jetzt mitten drin, beim Einlesen eine Errormeldung erscheint, so kann es sein, daß z.B. nur die Zeilenliste eingelesen werden konnte, vom Programmteil aber noch nichts. Es würde deshalb wenig Sinn ergeben, dies in irgendeiner Form zu retten.

CLUB-SEITE

Nachdem wir letztes Mal nur sehr wenig über unsere User-Clubs berichten konnten, haben wir heute mehr Stoff. Am Beginn steht eine geplante Neugründung:

Zur Gründung der 1. Bremerhavener TI 99/4A User Group sind alle interessierten Anwender in Bremerhaven und Umgebung aufgerufen, Kontakt mit mir aufzunehmen. Geplant sind wöchentliche Treffen und ein enger Kontakt zur 1. Bremer TI User Group, die ich auf diesem Weg herzlich grüße. Näheres erfahrt Ihr von:
Andreas Froberg,
Klopstockstraße 4,
2850 Bremerhaven

Der nächste Club, der sich heute vorstellen will, ist nicht nur auf einen Computertyp beschränkt:

Die Computer Club I.G. ist nicht auf einen bestimmten Computertyp festgelegt. Die Mitglieder besitzen Computer vom SX Spectrum, TI 99/4A, C64 bis hin zum Apricot F1 oder auch gar keinen Computer.
Im Clubmagazin "Alles klar?!" sind wenig Listings, dafür viele Tips und Tricks. Der Mitgliedsbeitrag beträgt 2,– DM für Schüler und 4,– DM im Monat für Erwachsene. Der Club veranstaltet diverse Treffs und Informationsveranstaltungen und gibt Unterricht in Basic und anderen Programmiersprachen. Am 6. und 7.9.86 will die Computer Club I.G. die 4. Computertage veranstalten. Da wird unter anderem eine Briefmarkenausstellung über die Entwicklung der Computer zu sehen sein. Nähere Auskunft über die Computer Club I.G. gibt (bitte Rückporto beilegen):
Knut Reuther,
Wunstotferstraße 35,
3075 Neustadt 1

Aus dem benachbarten Ausland kommt die Nachricht von einer Börse für TI 99/4A Benutzer:

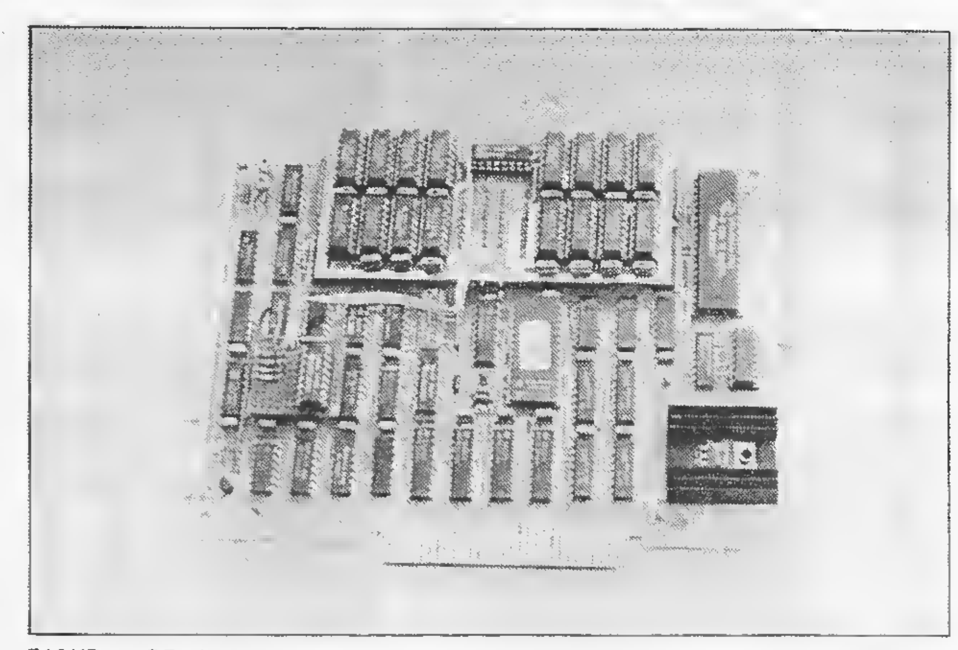
Am 17. Mai 1986 veranstaltet der nationale holländische Verein für TI 99/4A User eine Börse von 12 Uhr 30 bis 17 Uhr 30 in der "reehorst" in Ede in der Nähe von Arnheim. Es wird eine interessante Ausstellung der Aktivitäten der holländischen User. Auch werden verschiedene Lieferanten für den TI 99/4A Ihre Produkte anbieten. Der belgische Verein "Texsoft" wird auch repräsentiert sein. Dieser Tag ist bestimmt einen Ausflug wert.
Nähere Auskünfte bei:
Texas Instruments
Gebruikers Groep,
Zonegge 1406,
NL-6903 GH Zevenaar

Der Reigen heute beschließt wieder einmal der TI 99 Workshop Rheinland, der, wie man sieht, eine ganze Menge von Öffentlichkeitsarbeit versteht, aber das hatten wir ja schon einmal gesagt:

Am 1.3.1986 holte der TI 99er Workshop Rheinland zu seinem bisher größten Schlag in Sachen Öffentlichkeitsarbeit für den TI 99/4A aus. Im Rahmen eines Computerflohmarktes in Düsseldorf wurde erstmals die Demonstrationsshow "TI 99/4A in Perfektion" gezeigt. Aber zuerst etwas zur Vorgeschichte. Auf allen lokalen Computershows und -flohmärkten fiel uns das Fehlen des TI krass ins Auge. Durch Gespräche mit Usern anderer Systeme wurde auch sehr schnell deutlich, daß dort ein gerüttelt Maß an Unkenntnis über die Fähigkeiten unseres Systems vorherrschte. Und immer wieder die teilweise hämische Aufforderung, na dann zeigt doch mal was eure Kiste noch hergibt. Das wollten und konnten wir nicht auf uns sitzen lassen. Und schnell, wie wir nun mal sind, haben wir sofort die Veranstalter von Computershow, -messen und -flohmärkten angeschrieben, wann denn die nächsten Veranstaltungen wären. Nun, dabei kam uns die VHS Düsseldorf mit Ihrem Termin am 1. März sehr gelegen. Als der Termin feststand, die Tische gebucht waren und alles andere geklärt war, wurden die Kontakte zu verschiedenen Anbietern aktiviert, um von dort Werbematerial zu bekommen. Einige unserer Partner, die Fa. Reis, Fa. Atronic und Fa. Elektronik-Service, Kaarst, unterstützten uns dann auch sehr reichhaltig mit Preislisten und Produktbeschreibungen. An dieser Stelle nochmals vielen Dank dafür! So eingedeckt ging's am Morgen des 1. März nach Düsseldorf.
Als erstes wurden in der doch sehr großen Halle einmal unsere Tische gesucht. Und siehe da, wir hatten mal wieder das Glück des Tüchtigen. Unser Stand lag genau auf einer Empore gegenüber dem Haupteingang.
Als Ausstattung wurden zwei komplette Stationen aufgeboten, eine mit einem Farbfernseher, die andere mit einem Grün-Monitor.
Und schon beim Aufbau die ersten verwunderten Blicke; das sollte der Uralt-TI sein? Aber dann ging's erst richtig los!
Als die ersten Besucher die Halle betraten, wurde die erste Runde unserer Demo-Show eingeläutet. Es begann mit einigen hervorragenden Musikdemonstrationen aus der Feder von Roman Majer, dann folgten einige von uns geschriebene Demos der Apesoft Grafik-Befehle mit dem Extended Basic II +, danach wieder ein paar Musikstücke aus den USA.
Was sich dann an unserem Stand abspielte, war schier unglaublich! Die Besucher, angelockt von der lauten Musik, wollten mit dem Gedanken "na ja was soll's" schon wieder abdrehen, da sahen sie das etwa 4 Meter lange Transparent mit der Aufschrift "TI 99er Workshop Rheinland", und blieben erstaunt stehen. Das gibt's doch nicht, das ist doch nicht wahr und ähnliche Sprüche gab es zu Hauf.
Die Menschentraube um unseren Stand wuchs und wuchs, das Informationsmaterial wurde uns aus den Händen gerissen und die Leute konnten sich einfach nicht sattsehen. Fragen nach der Herkunft der Ausstattung wurden gestellt und immer wieder der Satz "Wenn ich gewußt hätte, daß es so etwas für den TI gibt, hätte ich nie verkauft, aber es hieß doch überall, es gibt nichts Neues mehr!"
Auch eine ganze Menge TI-Fans haben uns besucht. Bei ihnen war die Freude natürlich besonders groß, endlich mal wieder ihr System auf einer Computershow zu sehen.
Während der Demo-Shows wurden an dem zweiten Gerät Anwenderprogramme wie der TI-Writer, MS-Multiplan, aber auch Programmiersprachen wie FORTH und LOGO gezeigt und erklärt.
Auch einige Tips und Tricks aus der BASIC- und ASSEMBLER-Programmierung wurden gezeigt.

DIE SENSATION: 128 KB-GRAM-KARTE

Als ich für diesen Testbericht eine Platine für die Peripherie-Box und eine Anleitung bekam, traute ich kaum meinen Augen. Die neueste Entwicklung der Firma Mechatronic aus Sindelfingen nennt sich 128 bzw. 512 KByte GRAM-Karte. Die Karte wird ohne Gehäuse geliefert, wie es ja bei anderen Computern schon gang und gäbe ist, findet aber trotzdem sicheren Halt in der Peripherie-Box.
Nun stellt sich die Frage, was eine GRAM-Karte ist. Kurz gesagt, ist sie genau das Gegenstück zu den Modulen, welche hauptsächlich aus GROM's, also Nur-Lesespeichern, bestehen. Die 128 KB der kleinen Karte sind in zwei 64 KB-Bereiche aufgeteilt und bei der 512 KB-Version handelt es sich um 8 Banks je 64 KB. Die erste Bank ist als RAM geschaltet, wiederum aufgeteilt in einzelne Bank's im Adressbereich >6000 bis >5FFF. Sie ersetzt also keine Speichererweiterung, benötigt diese sogar zum Betrieb der eingebauten Hilfsprogramme. Die weiteren Banks sind als GRAM ausgeführt. Wie der Name schon sagt, als Schreib- und Lesespeicher in Art der GROM's. Die erste Bank ist zusätzlich auch als GRAM ansprechbar, auf einer anderen GROM-Basisadresse. Hier wird ein bisher weitgehend unbekannte Eigenheit des Betriebssystems des TI 99/4A ausgenutzt, die es erlaubt, mehrere Module, die nur GROM's beinhalten, parallel zu betreiben.
In jede dieser Banks der GRAM-Karte kann ein Modul untergebracht werden, bei manchen sogar zwei Module, je nachdem, wie die Speicher der beiden Module angeordnet sind.
Um ein Modul mit der GRAM-Karte emulieren zu können, muß man zu Anfang einmal das entsprechende Modul in den Modulport stekken und kann dieses dann mittels eines Hilfsprogramms der Karte auf Diskette abspeichern. Wenn dies geschehen ist, muß man nur noch eine Load-Datei schreiben, die nun der Karte sagt, welche Dateien alle von der Diskette geladen werden müssen, um ein bestimmtes Modul emulieren zu können. So ist das Abspeichern und wieder Laden von Modulen auf eine Diskette eine ganz einfache Sache.



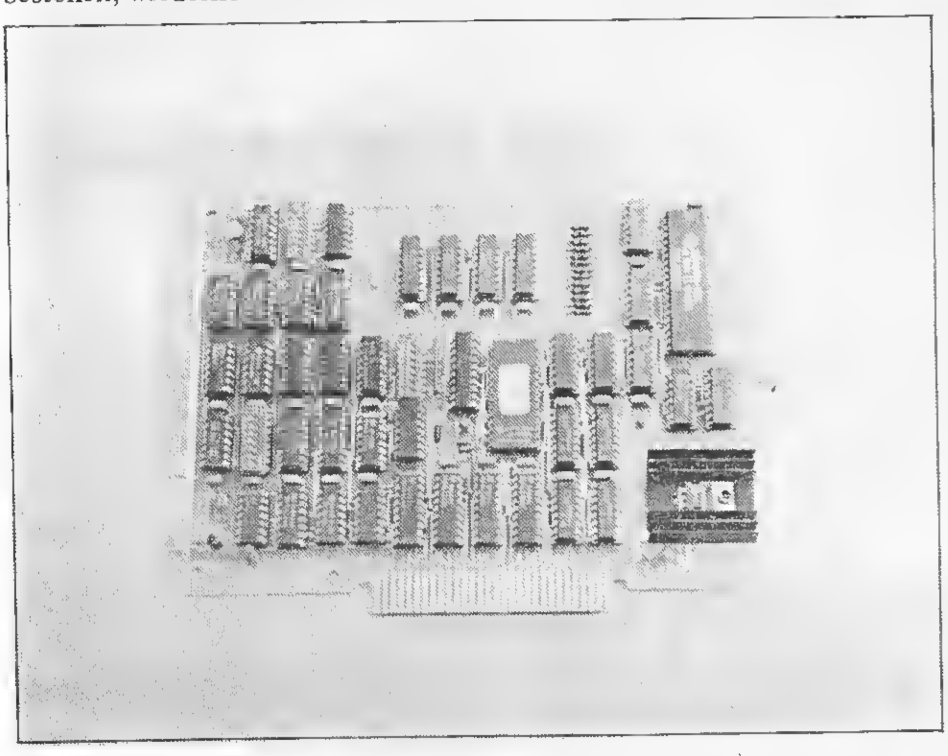
512KByte GRAM-Karte.

Das eben kurz angesprochene Abspeichern von Modulen stellt sich genauso als eine sehr einfache Arbeit dar. Zu diesem Zweck befindet sich in der Software, die in dieser Karte auf einem EPROM enthalten ist, ein Hilfsprogramm, das mit dem Befehl CALL GRAM aus dem Basic (oder Extended Basic) aufgerufen wird. Dabei kann ein beliebiges Modul zum Abspeichern eingesteckt sein. Dieses Hilfsprogramm meldet sich dann mit einem Bildschirm, aus dem die folgenden Unterroutinen mittels Tastendruck aufrufbar sind.

1 LOAD G(RAM) WITH PROGRAM
2 LOAD GRAM WITH ASM-FILE
3 LOAD RAM WITH ASM-FILE
4 LOAD GROM 0–2
5 SAVE GROM
6 SAVE ROM
7 LOAD FILE

Mit der ersten Routine kann man zuvor abgespeicherte Programme aus dem ROM oder dem GROM wieder in die GRAM-Karte einlesen und bearbeiten. Das Programm LOAD GRAM WITH ASM-FILE erlaubt es, selbstgestrickte GPL-Programme in die GRAM-Karte einzulesen. Um die nötigen ASM-Files erstellen zu können, benötigt man aber noch einen GPL-Assembler, der den richtigen Code erzeugt, wie z.B. der GPL-Assembler der Fa. Elektronik-Service.
Die RAM-Bank im Modulbereich kann mittels LOAD RAM WITH ASM-FILE für Assemblerprogramme genutzt werden. Dabei ist jedoch noch ein Hindernis vorhanden, denn man kann im "Modul" die Hilfs-Routinen des Assemblers (VSBW, VMBW usw.) nicht nutzen, sondern man muß erst neue Routinen dafür schreiben.
Mit LOAD GROM 0–2 können die GROM's der Konsole praktisch ausgeschaltet werden und man hat ein Duplikat davon in der GRAM-Karte, wo man sie dann problemlos manipulieren kann. Man kann zum Beispiel die Bildschirmfarben umändern, wenn man lieber Rot auf Schwarz oder anders schreiben möchte.
Allerdings weist die Anleitung zur GRAM-Karte ausdrücklich darauf hin, daß es bisher unklar ist, ob die GROMs der Konsole das verkraften, denn die GRAM-Karte arbeitet in diesem Modus gegen diese, aber bei mehreren 2-Stunden-Tests hat sich bei mir keine Auswirkung gezeigt. Damit ist aber noch nicht gesagt, daß es wirklich keine Schäden nach sich zieht. So sollte man es also nur dann längere Zeit benutzen, wenn

man noch einen Satz GROMs 0–2 zur Hand hat.
Mit SAVE GROM sind wir also bei der Abspeichern von Modulen angelangt. Wir müssen dann dem Computer nur noch sagen, welchen Bereich er auf der Diskette abspeichern soll. Da Module meist nicht nur aus einem GROM oder ROM bestehen, wiederholt sich die Prozedur mehrmals. Da wohl die meisten Anwender auch nicht wissen, aus wievielen GROMs und ROMs ein bestimmtes Modul besteht, ist in der Anleitung zur Karte eine ausführliche Liste der Module und ihrer Speicherelemente aufgeführt. Wie schon bei der vorherigen Routine gesagt, kann ein Modul auch ROMs enthalten oder ganz aus ROMs aufgebaut sein. Mit SAVE ROM kann man diese Banks dann auch noch mit abspeichern. Probleme kann es mit einigen Modulen der Firma Atari geben, denn dort sind die Längen der einzelnen ROM-Bänke unterschiedlich zu den Längen der ROMs aus TI-Modulen. Die Konstrukteure der GRAM-Karte mußten sich hier wohl entscheiden und haben die TI-Kompatibilität vorgezogen.
Jetzt kann man sagen, daß es doch ziemlich zeitaufwendig ist, wenn man ein Modul einladen möchte, es besteht ja auf der Diskette aus mehreren Dateien. Dazu eignet sich der Abschnitt LOAD FILE. Eine LOAD-Datei muß ganz einfach im DIS/VAR80 Format erzeugt werden und enthält in jedem Datensatz den Namen der zu ladenden Datei. Dann werden alle in der LOAD-File aufgeführten "Teile" des Moduls hintereinanderweg geladen.
Kommen wir aber nochmal zum Beginn zurück. Ist die GRAM-Karte in der P-Box eingesteckt und wird der Computer angeschaltet, benötigt er zur Initialisierung von 128 KB 4.5 Sekunden und bei der 512 KB-Karte beträgt diese Zeitspanne 18 Sekunden. Wenn die Initialisierung abgeschlossen ist, meldet sich der Computer mit dem normalen Anfangsbild. Wird eine Taste gedrückt, geht der Computer zum zweiten Bild, wo diese Auswahlliste erscheint:
1 FOR TI BASIC
2 FOR GRAMCARD >9800
3 FOR REVIEW MODULE LIBRARY
Man kann nun in das TI BASIC gehen oder, wenn man die 3 drückt, zur nächsten GRAMCARD >9820 bei der 128er und >9808 bei der 512er Version. Wenn man die 2 drückt, erscheint eine weitere Auswahlliste, welche normal die Anwahl des oben beschriebenen Hilfsprogramms, wie es auch mit CALL GRAM aufgerufen werden kann, oder das Aufrufen von diversen Load-Files erlaubt. Es stehen dabei 2*8 vordefinierte Namen zur Verfügung, mit so beziehungsreichen Bezeichnungen wie XB, EA, DM, . . . und eine Namenstabelle, die mit AA,BB anfängt und mit FF und GG endet. Durch Drükken der der gewünschten Lade-Datei entsprechendenen Taste wird diese dann von Diskette aufgerufen und eingelesen. Hierzu noch ein Trick, der nicht in der Anleitung steht: In der Load-Datei kann man durch Bestimmen der GROM-Basisadresse auch gleich die Bank bestimmen, in die das Modul eingelesen wird. Dazu muß nur hinter den Dateinamen in der Load-File nach einer Leerstelle die GROM-Basisadresse geschrieben werden, also z.B. 9800 oder 9820.
Soweit so gut, aber wie kann ich den Inhalt eines so abgespeicherten Moduls ändern, zum Beispiel den Namen oder die Bildschirmfarbe? Dazu gibt es zwei Möglichkeiten. Zum Ersten: Ich lese einen Debugger ein und kann dann den Speicher editieren, dies geht mit dem normalen TI-Debugger, oder, ich benütze den in die GRAM-Karte implementierten Editor. Dieser Editor ist zwar einfach gehalten, aber seine Anwendungsmöglichkeit ist dennoch nicht eingeschränkt, denn man kann mit ihm die drei Hauptspeicher des TI anschauen (CPU-RAM, VDP-RAM und das GRAM). Die Bedienung ist dabei bildschirmorientiert und denkbar einfach. Dieser kann ebenfalls aus dem Basic mit CALL MONITOR aufgerufen werden.



Die 128KByte GRAM-Karte.

Bis jetzt haben wir über das Module einlesen und abspeichern gesprochen, aber die GRAM-Karte kann noch mehr! Man kann ein TI-Basic-Programm in ein Modul umwandeln. Dies hat den Vorteil, daß für das Basic-Programm statt bisher ca. 13 KB ganze 26 KB RAM zur Verfügung steht. Dabei wird das reine Programm vom VDP-RAM in das GRAM umgeladen und das VDP-RAM steht voll für die Variablen zur Verfügung. An der Geschwindigkeit des Programmablaufs ändert sich nichts, obwohl sich der Interpreter des TI-Basic die Informationen von ganz woanders herholen muß. Das Programm wird dann nicht mehr mit RUN gestartet, sondern ist im Titelbild mit Namen aufrufbar.

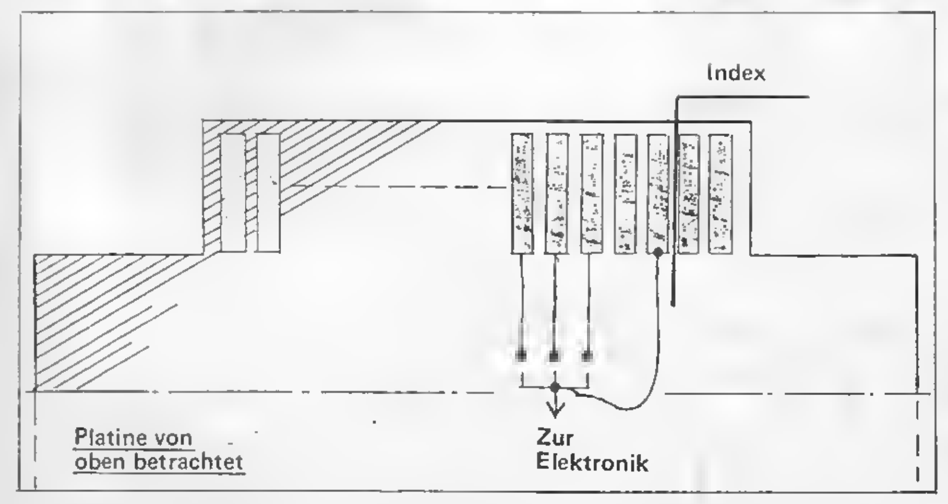
Abschließend möchte ich sagen, daß diese GRAM-Karte voll und ganz gelungen ist. Sie ist technisch einwandfrei und professionell aufgebaut. Dabei handelt es sich bei dieser Karte nicht um irgend einen TI-Nachbau, sondern die Konstrukteure haben genau gewußt, was sie wollten. Das Umwandeln der Basic-Programme in ein Modul und die Ausnutzung des Modul-Library

zeigen, daß hier Insider am Werk waren.
Alle Funktionen der GRAM-Karte sind Softwaregesteuert, es gibt also keine lästigen Schalter. Der Modulschacht wird überhaupt nicht mehr benötigt (es sei denn zum Auslesen von Modulen und für einige Spielmodule). Ist übrigens ein Modul eingesteckt, schaltet sich die GRAM-Karte automatisch ab. Mit dieser Karte eröffnen sich völlig neue Perspektiven für den TI 99/4A: Zusammen mit einem GPL-Assembler bietet sich hier ein Entwicklungssystem. Wer's kann, kann damit die Module seinen Wünschen entsprechend umprogrammieren.
Beim Betrieb der GRAM-Karte gab es keinerlei Schwierigkeiten im Zusammenspiel mit anderen Erweiterungen. Einzig beim CorComp-Diskkontroller muß mittels der Leertaste vom CorComp-Titelbild zum normalen TI-Titel zurückgekehrt werden. Es können bis zu 2 Stück 512KByte GRAM-Karten gleichzeitig in die Box gesteckt werden, was gesamt 1MByte GRAM ergibt. Ein Wermutstropfen aber bleibt – der Preis. Mit ca. 700,– DM für die 128KByte Version ist sie nicht ganz billig. Dafür erhält man dann ein wahres "IC-Grab".
Anzumerken bleibt noch, daß es sich bei der getesteten 128KByte-GRAM-Karte um ein Vorabmodell handelt, aber ich konnte mich davon überzeugen, daß die Serie in Qualität und Ausführung gleich ist. Bei der 512KByte Ausbaustufe war beim Muster noch eine handgestrickte Platine, was sich in der Serie natürlich ändern wird, aber hier kann von der Funktion her nicht mehr viel geändert werden. Dieser Ausbausatz wird dann wohl in Kürze erhältlich sein.

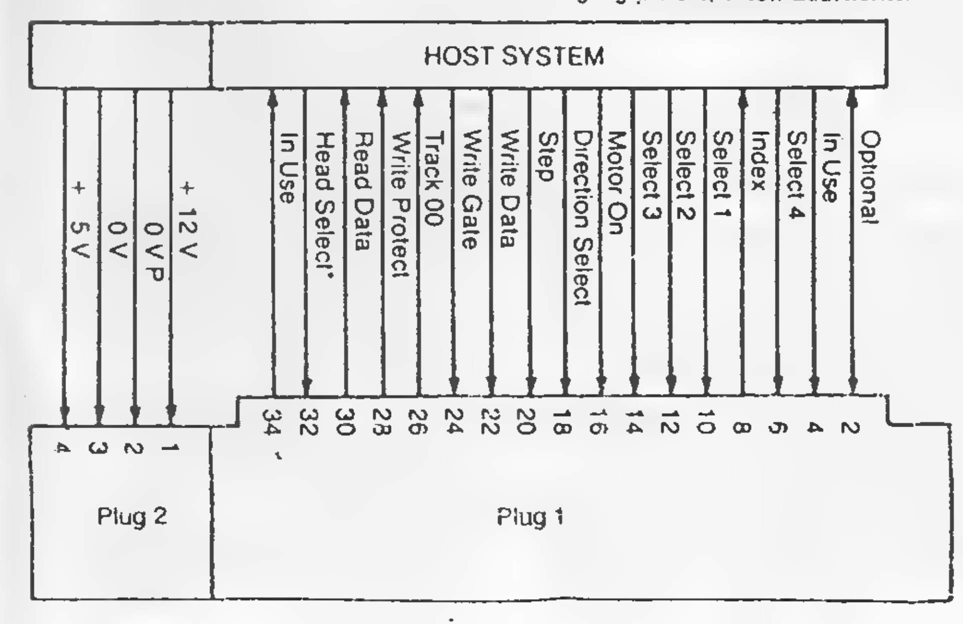
Hersteller: Mechatronic GmbH, Sindelfingen.
Bezugsquelle: Fachhandel

Heiko Liebald

ES GEHT AUCH MIT BILLIGLAUFWERKEN: DISK 4



Bei dieser Gelegenheit: Die genormte Anschlußbelegung der 5 1/4-Zoll Laufwerke:



Moderne Diskcontroller erlauben im Gegensatz zum TI-Controller auch ein viertes Laufwerk anzusprechen. Leider lassen sich jedoch die meisten preisgünstigen Laufwerke, obwohl ansonsten durchaus technisch identisch, nur als DSK1., DSK2. oder DSK3. codieren. Mit einem kleinen Kunstgriff läßt sich jedoch auch ein solches Laufwerk als Nr. 4 verwenden. Hierzu benötigen wir zwei Minuten Zeit, einen Lötkolben, eine ruhige Hand und zwei Zentimeter Draht.
Als erstes wird die Kodierbrücke, mit der das Laufwerk bisher bezeichnet wurde, entfernt. Danach legen wir das Laufwerk so auf den Tisch, daß die Steckerleiste zu uns herzeigt. Jetzt löten wir auf der Platinenoberseite unser kurzes Drahtstück zuerst an die Steckkontakte, und zwar von oben gesehen an den dritten Kontakt von links (in Schaltplänen als Nr. 6 bezeichnet). Unser Drahtstück löten wir jetzt mit dem anderen Drahtende an das zur Platineninnenseite zeigende Ende einer der drei Kodierbrücken. Kodierstecker **nicht** wieder aufsetzen. Fertig zum Testen.

K. Herdin

FÜR EINE HANDVOLL DOLLAR: TI-ARTIST-DER ZEICHENKÜNSTLER

TI-ARTIST

TI-ARTIST™

Draw D · Point P · LinE L · K-linE K · Rays R

Fill F · FramE V · Box X · CirclE O · Disc Q

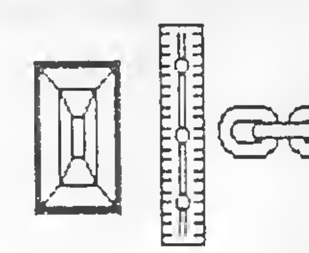
Clear Image ^A · H or V H · Swap N · InvErt I · Alpha Num E

Clear Color ^G · StorE S · Zoom Z · Mirror M · Hard Copy C

Plot/Erase · Color cursor · Pattern indicator · Foreground/Background

TI-AR
VERSION

INSCEBO



©1985 BY CHR

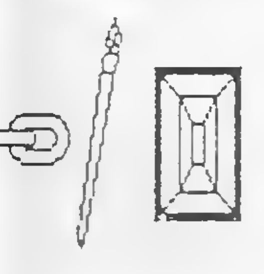
Für den kreativen Anwender des TI-99 wurden einige Zeichenprogramme entwickelt. Die in Deutschland wohl bekanntesten Programme für Kinder und Künstler sind der "Graphicmaster" und "3 D-World" von Wulf Becherer sowie das hardwareabhängige "Zeichenbrett".
Alle diese Programme haben eines gemeinsam: Sie waren relativ teuer (mittlerweile sind sie bereits auf einem annehmbaren Preis gesunken), sind teilweise sehr umständlich zu bedienen und lassen wichtige Funktionen vermissen.
Diese Mängel tragen natürlich nicht dazu bei, die Freude am Zeichnen zu fördern und auf lange Zeit zu konservieren, deshalb finden diese Zeichenprogramme auch relativ schnell ihren Weg ins große Programmarchiv und dienen nur noch statistischen Zwecken.
Mit ganz anderen Augen sehen wir ein Zeichenprogramm, das letztes Jahr in den USA entwickelt wurde. Dem Autor, Chris Faherty, gelang es, ein Programm für den TI zu entwickeln, das alle anderen Programme dieser Art weit in den Schatten stellt. TI-Artist wurde geboren und für ein langes kreatives Leben voller Zeichenfreude ausgerüstet. Das komplett in Assembler geschriebene Programm füllt bis auf 5 Sektoren eine ganze Diskette (SS/SD).
Um es einem möglichst großen Anwenderkreis zugänglich zu machen, wurden mehrere Startroutinen integriert, die es auf verschiedenen Konfigurationen lauffähig macht. Vorausgesetzt werden lediglich 1 Diskettenlaufwerk und zumindest die 32K-Erweiterung. Welches Modul der Anwender einsetzt, bleibt zum größten Teil ihm selbst überlassen. Die 20seitige Dokumentation erklärt genaustens den Ladevorgang von TI-Artist mit dem Assemblermodul, Mini Memory, Extended Basic, dem TI-Writer, CorComp-Manager und dem Myarc Diskcontroller.
Auf der Diskette findet sich sogar ein Installationsprogramm, um in einer 128K-Erweiterung eine RAM-

Disk für TI-Artist anzulegen. Der Zugriff auf die einzelnen Programmteile wird dadurch ungleich schneller, als wenn sie jedesmal von Diskette nachgeladen werden müssen. Als Zeichenutensilien sind Joystick, Tastatur und Trackball vorgesehen. Durch die offene und im Handbuch gut erklärte DSR-Routine kann der erfahrene Anwender weitere Eingabegeräte programmieren.

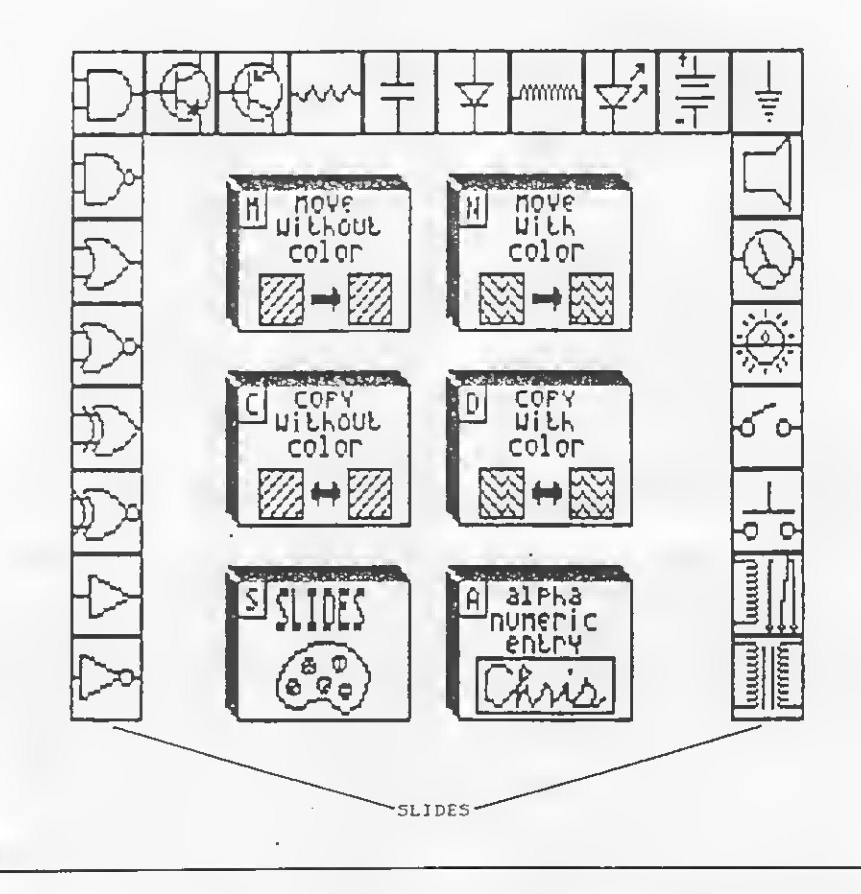
ein Druck auf die Feuertaste wandelt das "F" in ein "B" um und das Programm weiß, daß die nächste angewählte Farbe als Hintergrund gedacht ist. Als zusätzliche Hilfe nimmt das Fadenkreuz des Cursors noch die Farbe des Vordergrunds an und ein Pixel in der linken oberen Ecke des Cursorfeldes zeigt die angewählte Hintergrundfarbe. Jede Funktion kann auf zwei ver-

kann fast gänzlich auf das Menü verzichten, da auch im Zeichenmodus die verschiedenen Funktionen durch ihr alphanumerisches Äquivalent aufgerufen werden können. So dient der Punkt zum Beispiel zum Umschalten zwischen Pinsel und Radiergummi, das Semikolon regelt die Geschwindigkeit des Cursors und Farben werden mit dem Komma auf der Tastatur ausgewählt. Der

TIST
2.0
T, INC.



S FAHERTY



Insgesamt besteht der Zeichenkünstler aus vier Hauptteilen.

1.) Dem eigentlichen Malprogramm, TI-Artist:
Nach Anwahl dieses Programmteils wird der Benutzer mit einer Menütechnik konfrontiert, die an Übersichtlichkeit und Anwenderfreundlichkeit nicht das geringste zu wünschen übrig läßt.
20 dreidimensionale Blöcke beinhalten die verschiedenen Arten, in denen gezeichnet werden kann. Darunter befinden sich acht weitere Boxen zur Auswahl der Pinselart und -stärke. Ein Kästchen mit einem stilisierten Visier wandelt den Pinsel in einen Radiergummi um. Die letzte Reihe enthält alle 16 Farben, die als Vorder- und Hintergrundfarbe benutzt werden können, einen Farbcursor, vorprogrammierte Patterns zum Ausfüllen einer Fläche und natürlich eine Umschaltmöglichkeit zwischen Vorder- und Hintergrundfarbe.
Hierzu wird der Cursor einfach über das Kästchen mit dem "F" geführt,

schiedene Arten aufgerufen werden. Zum einen durch Anklicken mit Joystick oder Maus und zum anderen durch Eingabe des Kennbuchstabens der Funktion. Dies bedeutet bei TI-Artist einen nicht zu unterschätzenden Vorteil!
Das Menü nimmt einen kompletten Bildschirm ein, um in den Zeichenmodus zu schalten wird nur kurz die Leertaste niedergehalten.
Dank dieser Technik ist wirklich der ganze Bildschirm zum Zeichnen frei und der fortgeschrittene Benutzer

Einsteiger gelangt jederzeit mit Hilfe der Spacetaste ins Hauptmenü zurück, um dort leichter einzelne Optionen anzuwählen.
Das Hauptmenü ist derart reichhaltig ausgestattet, daß wir eigentlich nur zwei Funktionen vermißten. Ein Menüpunkt zum einfachen und genauen Zeichnen von Ellipsen und ein anderer für die Generation von Halbkreisen. Ansonsten ist alles, aber auch wirklich alles enthalten. Freihandzeichnen, Kreise, Boxen, verschiedene Arten von Linien,

Fill- und Zoomfunktionen sowie die Möglichkeit, Farben einzelner Bildteile schnell auszuwechseln oder zu löschen, um nur einige Beispiele aus dieser Palette zu nennen.
Ein Menüpunkt dient zur Eingabe von alphanumerischen Zeichen, um zum Beispiel ein Bild zu beschriften oder ein Logo für Briefpapier aufzubauen. Die standardmäßig im ASCII-Code enthaltenen Schriftzeichen können in 9 verschiedenen Höhen und 9 verschiedenen Breiten beliebig kombiniert werden. Dem Anwender stehen insgesamt also 81 Schriftgrößen zur Verfügung.
Zwei weitere Menüpunkte rufen Untermenüs auf, die genauso anwenderfreundlich wie das Hauptprogramm das Laden, Abspeichern und Ausdrucken bestehender Bilder übernehmen. Auch hier ist es dem Anwender freigestellt, ob er die einzelnen Optionen mit dem Joystick ansteuern will oder lieber die entsprechende Taste drückt.
Für die Hardcopy-Routine wurden bereits einige Druckertypen voreingestellt, die nur noch aufgerufen werden müssen. Da unser Drucker Epson-kompatibel ist, lag natürlich nichts näher, als im Menü auch "Epson" auszuwählen.
Im Dialog klärte TI-Artist dann mit uns, wie der Ausdruck auszusehen hatte. Von der 1:1-Darstellung in einfacher Dichte bis zur dreifachen Vergrößerung und dreifachen Dichte war alles erlaubt und funktionierte einwandfrei. Es war eine reine Freude, den Drucker bei seiner schnellen und korrekten Arbeit zu beobachten.
Store, wie der Name schon verrät, dient zum Abspeichern neuer Bilder und Laden bereits bestehender Zeichnungen. Je mehr Bilder bereits auf Diskette gespeichert wurden, desto schwieriger fällt es, die Namen auseinander zu halten. Bei TI-Artist kein Problem. Der Zeichenkünstler erstellt auf Wunsch ein Inhaltsverzeichnis aller Bilder auf Diskette. Das Abspeicherungsformat der Zeichnungen (Dis/Var80) wird im Handbuch bestens erklärt. Die Bilder werden im dezimalen Zahlensystem, wie im Druckerhandbuch unter "Bit Map Modus" beschrieben, abgespeichert. Sie können mit ein wenig Erfahrung in eigene Programme übernommen werden und sind durch ihr Format auch für den TI-Writer und den Editor des Assemblermoduls les- und modifizierbar.
Für die großartigen Möglichkeiten, die sich dadurch ergeben, hätte man sich für den Nichtfachmann in der Begleitdokumentation zumindest ein konkretes Beispiel gewünscht. So beschränkt sich das Handbuch aber auf die detaillierte Beschreibung der Abspeicherungsart und überläßt die eigentliche Umwandlung dem Können des Anwenders oder dem Einfallsreichtum des Laien.
Befindet man sich beim Abspeichern eines Bildes gerade im Zoom-Feld, wird nur dieses Fenster aus der Zeichnung herausgepickt und auf Diskette überschrieben.
Das letzte Untermenü TI-Artist's ist über eine etwas ungewöhnliche und gefürchtete Tastenkombination des TI 99 zu erreichen, FCTN QUIT. Mancher TI-User kann ein Lied davon singen, was passiert, wenn er stundenlang ein Listing abgetippt hat und dann bei bereits nachlassender Konzentration anstatt der "="-Taste die Quit-Taste erwischte. Doch keine Angst, bei TI-Artist erscheint weder das Titelbild, noch wird irgendein Teil der Zeichnung gelöscht.
Durch diese Kombination wird lediglich ein weiteres Menü als Fortsetzung des Hauptmenüs aufgerufen, dessen Funktionen allerdings weniger häufiger benötigt werden und der Übersicht und des Platzbedarfes wegen, ausgelagert werden konnte.
In diesem Teil TI-Artist's ist es möglich, beliebig große Teile der Zeichnung mit oder ohne Farben an eine andere Stelle des Bildschirms zu versetzen oder zu kopieren, wobei der Unterschied dieser zwei Möglichkeiten nur darin besteht, daß bei der ersten Funktion das alte Fenster gelöscht wird und im zweiten Fall erhalten bleibt.
Umrandet werden die sechs neuen Auswahlmöglichkeiten durch 24 Zeichen der Elektrotechnik, die nichts anderes darstellen als 24 umdefinierte Zeichen des ASCII-Codes. Diese Symbole dürfen nicht nur in der eigenen Zeichnung verwendet werden, sondern lassen sich auch nach allen Richtungen drehen, vervielfältigen oder durch eigene Symbole ersetzen. Selbstverständlich können auch die neu erstellten Charakter auf Diskette abgespeichert und bei Bedarf wieder aufgerufen werden.
Genauso leicht lassen sich beliebig große Bildausschnitte umranden, seperat abspeichern und an andere Zeichnungen anfügen.
Dateinamen läßt TI-Artist nur bis acht Buchstaben zu. Die zwei verbleibenden Zeichen werden vom Programm selbständig angehängt. Als sogenannte "Extensions", also Erweiterungen kennzeichnet TI-Artist komplette Bilder, Ausschnitte, umdefinierte Charakter oder neu erstellte Schrifttypen jeweils mit einem anderen Anhängsel, ohne daß sich der Anwender darum zu kümmern braucht. Dementsprechend ist das Disketteninhaltsverzeichnis auf dem Bildschirm auch wie eine Tabelle untergliedert und listet die verschiedenen Bestandteile unter der jeweils richtigen Kategorie auf.
Benutzerfreundlichkeit total und Ausnutzung aller Möglichkeiten, die der TI 99 zur Verfügung stellt, war das Motto von Chris Faherty, dem Autoren dieses Programms. Mancher Kollege kann sich bei ihm eine deftige Scheibe abschneiden, was Können und laiengerechtes Umsetzen dieser Fähigkeit betrifft. Das Handbuch dient praktisch nur zur kurzen Vorabinformation und kann nach den ersten Schritten bereits für immer im Archiv verschwinden. Die Menütechnik ist "narrensicher" und genügt voll und ganz zur Anwendung des Zeichenprogramms. Zudem ist TI-Artist, als unabdingbare Voraussetzung an ein professionelles Computerprogramm, auch vollkommen absturzsicher. Kein noch so dummer Fehler läßt die Daten oder Bilder auf Nimmerwiedersehen verschwinden. Die einzige Möglichkeit, Teile zu löschen, besteht entweder gewollt durch Anwahl des entsprechenden Menüpunktes oder durch Betätigen des Ausschalters.

Programm: TI-ARTIST V2.0
Preis: 20 US-Dollar
Programmiersprache: TMS9900 Assembler
Bestelladresse: INSCEBOT INC.
Chris Faherty
P.O Box 260
Arnold, MD 21012

Wir konnten Ihnen im Rahmen dieses Testberichtes selbstverständlich nur einen kleinen Einblick in die wesentlichen Optionen des Zeichenkünstlers geben. Bei diesem Programm heißt es: "Selbst ausprobieren", doch leider ist es bis jetzt bei keinem deutschsprachigem Anbieter für TI Hard- oder Software zu erhalten.
Einzige derzeitige Möglichkeit, in den Genuß dieses Superprogrammes zu kommen, ist der direkte Kontakt zu dem Autor über den großen Teich. Findet er neben der Bestellung noch eine 20 Dollarnote, quasi als kleine Anerkennung seiner hervorragenden Leistungen, denn mehr ist es ja nicht, ist es eigentlich nur noch eine Frage des langen Transportweges, bis Sie sich so richtig kreativ mit diesem Programm austoben können.

PLÄDOYER FÜR EINEN COMPUTER, DER AUCH HEUTE NOCH MODERN IST

Der eingefleischte TI 99/4A-User hat es im Gespräch mit anderen Computer-Besitzern immer wieder schwer, die Fähigkeiten seines TI 99/4A gegen "Argumente" wie "was willst Du denn mit deiner alten Kiste" zu verteidigen. Diverse Tele-



fongespräche mit unseren Lesern und auch ein Leserbrief sind denn auch der Aufhänger für diesen Bericht. Es mag sein, daß die folgenden Zeilen etwas einseitig klingen, aber das, was andere Leute heutzutage in der Computertechnik, pardon, ich muß schon beginnen, in der Heimcomputertechnik für den letzten Schrei halten, muß wohl doch einmal mit einer gesunden Portion Skepsis betrachtet werden.
Natürlich muß der schon erwähnte Leserbrief diesem Artikel vorangestellt werden:
„Seit längerer Zeit beobachte ich nun schon den Zubehörmarkt für den TI 99/4A, der ja gerade in den letzten Monaten durch Neuentwicklungen und bessere Software erweitert worden ist. Zum Glück für viele, die ihren 99er erst kurz vor dem Totalausverkauf erstanden haben und seither auf der Suche nach Peripherie fast verzweifelt sind, und natürlich auch diejenigen, die ihr System erst als komplett betrachten, wenn sie in den Anzeigen der Fachzeitschriften nichts Neues mehr finden. Aber anscheinend sind gerade unter den wenigen Händlern, die noch oder wieder Peripherie und Software für den TI 99 anbieten, die meisten entweder clever oder mies, je nach der Sicht des Betrachters, wobei für mich jedenfalls letzteres mehr in Frage kommt.
Bei diesen Preisen, wie sie für den TI geradezu "charakteristisch" sind, liegt die Vermutung nahe, daß hier eine "Dreiviertel-Monopolstellung" und somit auch der Käufer einfach ausgenutzt wird.
Die billigste ("richtige") Diskettenstation ist nur zusammen mit Schnittstellen und RAM-Erweiterung für ganze 1600,– DM, die externe 32K-Erweiterung für nur 280,– DM und schließlich als krönender Abschluß die TI-Maus für einen Sparpreis von 300,– DM zu haben. Häufigste Begründung: "Ist ja immer noch billiger als die

Originalhardware von Texas Instruments." Dazu kann ich nur sagen: "Einfach lächerlich". Den besten Beweis hierfür hat die TI-REVUE mit ihrer 75,– DM Speichererweiterung geliefert, wobei von diesem Preis auf Grund der Herstellungszahl und des Direkteinkaufs bei der professionellen Herstellung noch einmal ein großer Teil wegfällt. Ähnliches zeigt ein einfacher Preisvergleich zwischen dem "Veteran" und dem "Neuling": Zahlt man für einen TI 99/4A + CPS 99 + Logo + XBas + Maus rund 2700,– DM, so bekommt man für den gleichen Preis den neuen Atari 260ST mit Monitor, mit Diskstation, Maus, 512KB RAM, 80 Zeichen/Zeile, 68000 CPU . . . Dies soll nicht heißen, daß der 99er ein schlechter Computer oder eine alte Rumpelkiste ist, aber die Technik hat sich verbessert und ist billiger geworden, mit einer Einschränkung: Dem TI 99.
Bei dieser Preispolitik habe ich die Befürchtung, daß mehr und mehr User ihre TI verkaufen oder in einer Schrankecke verstauben lassen und auf andere Systeme umsteigen, bei denen, wie man so schön sagt, Preis und Leistung stimmen. Deshalb sollte sich die Industrie diesen Sachverhalt noch genauer durch den Kopf gehen lassen, will sie nicht über kurz oder lang die Produktion für TI-Peripherie einstellen müssen. Sätze wie "wir lassen den User nicht im Stich" oder "sehr erfreulich ist die Lage auf dem Zubehörsektor zu beurteilen . . . da ja die Preise unter denen der Originalgeräte liegen" (TI-REVUE 9/85) sind für beide Seiten vorsichtig zu genießen.
Schließlich hat der TI 99/4A ja auch einmal 1300,– DM gekostet. Aber während dieser nachher für 300,– DM, also rund 1/4 des ursprünglichen Preises ausverkauft wurde, behielt die Peripherie ihre Preise bei." *Jürgen Günther, Obersulm*

So weit also der Leserbrief. Dabei soll hier gar nicht bestritten werden, daß es auch beim Zubehör für den TI 99/4A Preisgestaltungen gibt, die am Rande des "Erträglichen" liegen, aber offensichtlich ist sich Jürgen Günther nicht ganz darüber im Klaren, wieviel Entwicklungskosten es bedarf, um neue Geräte für den TI 99/4A auf den Markt zu bringen, denn eine wie auch immer geartetete Unterstützung seitens Texas Instruments gibt es nicht. Und dabei sind natürlich dann auch immer die verkauften Stückzahlen zu rechnen, denn Entwicklungskosten von mehreren zigtausend Mark sind schnell beieinander. Und wir müssen uns wohl damit abfinden, daß der Markt langsam kleiner wird. So gehört wohl heute durchaus unternehmerischer Mut dazu, noch neue Produkte für den TI 99/4A auf den Markt zu bringen, denn verkaufte Stückzahlen von nur ein paar hundert Stück sind durchaus nichts Außergewöhnliches.
Nun soll hier natürlich auch nicht irgendeiner Preistreiberei das Wort geredet werden. Unrühmliche Ausnahmen gibt es überall. Wer aber den Markt für TI-Zubehör in der letzten Zeit beobachtet hat, sieht ja wohl, daß die Gesetze von Angebot und Nachfrage noch nicht außer Kraft gesetzt sind. Preissenkungen sind durchaus immer noch zu sehen, wobei wir aber damit wohl auch wieder bei den Stückzahlen sind.
Wir wollen hier aber nicht in erster Linie von Preisen reden, sondern vom sogenannten "Stand der Technik". Wenn hier der Atari ST als "Stand der Technik" hingestellt wird, so entbehrt das allerdings nicht einer gewissen Komik. Er könnte das ja vielleicht mal werden, wenn so alles vollständig entwickelt ist, bis dahin aber dürfte er dazu wiederum zu alt sein. Immerhin gibt es ja mittlerweile eine Version des Betriebssystems (genannt TOS, bei boshaften Leuten auch schon Trouble Operating System genannt), welche halbwegs die Anforderungen, die ein TI-User an Fehlerfreiheit stellen muß, erfüllt, aber eben nur halbwegs.
Ich kenne da eine Familie, deren Söhne noch vor ein paar Wochen darin wetteiferten, wer die meisten Wölkchen auf den Bildschirm bei einem Systemabsturz produzieren konnte. Und das Basic des Atari ST scheint auch nicht gerade durch sorgfältige Entwicklung zu glänzen. Immerhin gibt es jetzt schon eine Zeitschrift, die zu einem Wettbewerb aufgerufen hat, wer die schönsten Fehler in dem Basic findet. Dabei wurden dann auch gleich eine ganze Menge Fehler veröffentlicht. Da kommt nun sicher gleich der Einwand "das kommt ja alles". Sicherlich, aber von Texas Instruments und auch von vielen der heutigen Zubehörlieferanten für den TI 99/4A bin ich anderes gewöhnt, was halt leider auch seinen Preis hat.

Kommen wir zum nächsten Argument, 512KByte RAM. Nun, wenn das Basic ohne Tricks geladen wird, bleibt davon nur sehr wenig übrig. Da habe ich auf meinem TI im Basic mehr freien Speicherplatz. Speicherplatz alleine ist keine Hexerei, wenn auch in der heutigen RAM-Verrücktheit ein wichtiges Verkaufsargument. Dies aber wohl auch nur im Heimcomputer-Bereich, denn bei den mehr professionellen PCs reichen heute 512KB durchaus für die tollsten Anwenderprogramme. Nur, Speicher muß ich auch nutzen können, und da hapert es dann doch manchmal, zugegebenermaßen auch beim TI, wenn ich da schon an einige Module denke, die nicht einmal die Speichererweiterung benutzen können. Rein formal ist das aber auch da kein Problem: Wir hatten in der TI-REVUE schon einmal eine Rechnung. Und mittlerweile besitzt einer der Redaktions-TI 99/4A um die 1,8M-Byte RAM, mit dem neuen Mega-Ram von Atronic sind das dann nochmal 512KByte mehr.
Wenden wir uns aber wieder seriöseren Dingen zu, weil nicht nur vom Geschmack oder der Werbung beeinflußt: Dem Betriebssystem. Warum denn kann der TI 99/4A auch heute noch jeder Neuerung angepaßt werden? Doch nur deshalb, weil sich Texas Instruments eine ganze Menge überlegt hat. Das Geheimnis liegt über das DSRLNK im Betriebssystem. Hiermit kann ich bis zu 16 Peripheriegeräte, die ihr eigenes Betriebssystem haben, problemlos in das ganze System einbinden. Einzige Bedingung ist, daß sich das Betriebssystem des Peripheriegerätes an gewisse "Spielregeln" hält.
Dann ist da ja auch noch das Betriebssystem der Peripheriegeräte. Nehmen wir als Beispiel das DOS, also das Betriebssystem für Diskettenlaufwerke. Der TI beherrscht hier ja von Anfang an relative Dateien, also Dateien mit wahlfreiem Zugriff auf jeden Datensatz. Das war und ist bei weitem noch immer nicht Standard im Heimcomputermarkt. Denn wenn ich eine relative Datei erst einmal sequentiell schreiben muß, also alle Datensätze nacheinander in der richtigen Reihenfolge, wie beim Atari, bevor ich auf diese Datei relativ zugreifen kann, so ist das für mich nicht als Stand der Technik zu bezeichnen. Auch die Rechengenauigkeit des TI 99/4A ist bis heute nur von wenigen, meist viel teureren Computern übertroffen (nicht die Anzahl der Stellen beachten, sondern nachsehen, ab welcher Stelle das Ergebnis nicht mehr stimmt!). Bleibt noch die Auslegung der Schaltung. Texas Instruments hat hier sehr aufwendig gearbeitet. An dieses

Merkmal müssen sich nun teilweise auch die Fremdhersteller halten. Die Vorteile liegen klar auf der Hand: Höhere Datensicherheit und bessere Funktionstüchtigkeit. Der Nachteil auch: So etwas ist natürlich teurer. Wer nicht glaubt, daß es um die Funktionstüchtigkeit der heutigen Computer nicht mehr so gut bestellt ist, der gehe einmal zu einem Fachhändler und frage ihn, wieviel Prozent der Geräte aus der Packung heraus nicht funktionieren. Vielleicht gibt dieser sogar eine der Wahrheit entsprechende Antwort.

Fazit: Der TI 99/4A hat, wie jedes technische Gerät, seine Vorzüge und auch seine Nachteile. Letztere sind in der Tastatur und in der Ausführungsgeschwindigkeit hinreichend bekannt (hinsichtlich der 80 Zeichen und einer verbesserten Grafik konnte ich neulich so ein kleines Kästchen sehen, so daß diesem Manko wohl noch in diesem Jahr abgeholfen wird). Der TI 99/4A aber kann heute noch in manchen Bereichen Maßstäbe setzen, er war eben seiner Zeit doch etwas voraus. Zu nennen ist hier das Extended Basic und auch das Betriebssystem wie der Aufbau.

Heiner Martin

NEUES FÜR DEN TI: MEGA RAM VON ALTRONIK

Unter der Bezeichnung MEGA-RAM hat die Hamburger Firma Atronic einen 256 K-Byte großen Speicher für den TI 99/4A angekündigt, der bis 1 MByte ausbaubar ist. In der Grundversion stehen 8 RAM-Blöcke mit jeweils 32K-Byte zur Verfügung. Zusätzlich ist eine Schnittstelle für sogenannte Softcards eingebaut. Die sind Speicher-Chips in Scheckkarten-Format und in Zukunft soll darauf auch Software für den TI 99/4A angeboten werden. Die im MEGA-RAM fest eingebaute Software läßt sich in zwei Bereiche aufteilen: Einmal ist die Software für die Nutzung des MEGA-RAM's als RAM-Disk vorhanden. Der Speicher läßt sich damit praktisch wie eine Diskette nutzen, nur viel schneller. Zur RAM-Disk-Software gehört auch ein RAM-Disk Manager, der auch einen vorhandenen Disk-Controller unterstützt.

Weiter gibt es ein als SUPERVISIOR bezeichnetes Programm. Es kontrolliert die Funktion des RAM-Disk Managers und organisiert die Verteilung der RAM-Blöcke.

Wir werden über das MEGA-RAM sobald wie möglich einen Testbericht veröffentlichen.

ASSEMBLER: DIE CRU

Wie schon in der letzten Folge angekündigt, wollen wir uns in Zukunft nicht mehr in dem engen Rahmen eines Assembler-Lehrgangs mit der Maschinensprache des TI 99/4A beschäftigen, sondern gezielter auf bestimmte Probleme eingehen.

Zu den häufig mißverstandenen Eigenschaften des TMS 9900 gehört ganz sicher die Communications

Register Unit, kurz auch als CRU bezeichnet. Bei dieser Kurzform wollen wir es in Zukunft auch belassen. Der TMS 9900 besitzt ja Adressleitungen und Datenleitungen. Bei den „normalen" Assemblerbefehlen wie MOV und JMP usw. wird nun jeweils über die Adressleitungen das Byte (oder Wort) angewählt, welches vom Prozessor benötigt wird. Über die Datenleitungen wird dann der Wert aus dem Speicher gelesen bzw. hineingeschrieben. Um der ganzen Hardware noch zusätzliche Informationen zu geben, zeigt die Leitung MEMEN einen derartigen Speicher-Zugriff an. DBIN sagt, in welche Richtung die Daten übertragen werden und WE gibt bei einem Schreibvorgang die entsprechenden Hinweise, wann die Daten vom Speicher zu übernehmen sind. Um die CRU besser verstehen zu können, betrachten wir diese erst einmal völlig unabhängig von diesem normalen Speicherzugriff, denn damit hat sie auch gar nichts zu tun! Auch die CRU ist prinzipiell ein Speicherzugriff, nur besitzt hier der Speicher nicht die Breite eines Wortes (16 Bit) oder Bytes (8 Bit) sondern es ist nur 1 Bit auf jeder Adresse vorhanden. Daraus folgert, daß ich nur eine Datenleitung brauche, Adressleitungen aber wie gehabt. Die Datenleitung ist bei der CRU aber noch getrennt in Datenleitung zum Lesen (CRUIN) und Datenleitung zum Schreiben (CRUOUT). Beim Schreiben der Daten ist natürlich für den Speicher wieder der Zeitpunkt wichtig, wann die anliegenden Daten in den Speicher übernommen werden sollen. Dies erledigt ein Impuls auf der Leitung CRUCLK (CRU Clock).
Nun wäre es natürlich unsinnig, für die ganzen CRU-Speicherstellen nochmal getrennte Adressleitungen herauszuführen. Diese Aufgabe übernehmen beim TMS 9900 die ganz normalen Adressleitungen. Hier müssen wir daran denken, daß am Prozessor nur die Leitungen A0 bis A14 zur Verfügung stehen (16 Bit-Prozessor), A15 wird durch eine besondere Logik im Rechner für den I/O-Port zusätzlich erzeugt. Auf dieses A15 können wir aber dann auch CRUOUT legen, denn A 15 wird ja eigentlich bei einem CRU-Zugriff nicht benötigt. Weiter ist der Prozessor bei einem CRU-Zugriff nicht in der Lage, die Adressleitungen A0 bis A2 zu steuern, so ergibt sich ein CRU-Adressumfang von >0000 bis >1FFE (nur gerade Adressen, also insgesamt 4096 Bits).
Die Struktur dieser CRU deutet schon darauf hin, daß dieser Speicher mehr für direkte Schaltvorgänge oder Abfragen einzelner Bits (Schalteingänge) geeignet ist, und dafür wird die CRU im TI 99/4A auch hauptsächlich verwendet. Die Speicher sind dabei im allgemeinen einfache Flip-Flops, die ja dann den entsprechenden Wert speichern. In der Konsole findet der Adressbereich >0000 bis >0FFE Verwendung, es ist darauf aber nur ein TMS 9901 angesteuert, der neben dem Interrupt-Handling die Tastaturabfrage und Kassettensteuerung erledigt. Interessanter ist erst einmal der für Peripheriegeräte reservierte CRU-Adressenbereich von >1000 bis >1F00. Texas Instruments hat hier die Festlegung getroffen, daß jeweils um >0100 steigend die CRU-Basisadresse für Peripheriegeräte liegt. Wird dieses CRU-Bit jeweils eingeschaltet, so ist das Gerät bzw. das Eprom mit der Steuersoftware eingeschaltet. Üblicherweise wird dies immer durch die Hilfsroutine DSRLNK erledigt.
Jetzt haben wir uns erst einmal lange genug bei der Theorie aufgehalten. Wie kann nun der Prozessor mit der CRU arbeiten? Da, wie erläutert, die CRU einen eigenen Adressenzähler braucht, steht dem TMS 9900 das normale Adressregister nicht zur Verfügung. Für die CRU übernimmt dies das normale Register 12. Vor jedem CRU-Befehl müssen wir also R12 mit der richtigen CRU-Adresse laden. Über SBO &Set bit one können wir dann das Bit setzen und mittels SBZ (Set bit Zero) wieder löschen. Mittels TB (Test bit) wird das CRU-Bit in das Equal-Bit des Statusregister eingelesen. Dabei kann bei diesen Befehlen ein Offset angegeben werden. d.h. eine Differenz zur Adresse in R 12. Aber Achtung. das niedrigwertigste Bit in R 12 zählt nicht zur Adresse (wie oben geschildert, nur A3 bis A14 zählt!). Die folgenden beiden Beispiele haben daher die gleiche Wirkung:

```
 LI    R12,>1100
 SBO   1
und
 LI    R12,>1102
 SBO   0
```

Diese Eigenschaft, daß, um das nächste CRU-Bit zu erreichen, R12 um 2 erhöht werden muß, ist sehr wichtig. Sie muß immer bei den einzelnen Operationen berücksichtigt werden. Übrigens ist in dem Beispiel eben die CRU-Adresse die des Disk-Controllers und mit SBO 1 wird der Motor angeschaltet. Genauer schaltet die ansteigende Flanke des CRU-Ausgangsbit ein Monoflop, welches den Laufwerkmotor eine gewisse Zeit anschaltet.
Nun stehen für die CRU aber nicht nur Bit-Befehle zur Verfügung, sondern auch welche, mit der sich gleichzeitig mehrere Bits ausgeben bzw. einlesen lassen. Diese werden dann ausgehend von der in R12 vorhandenen Basisadresse immer in die folgenden höheren Bits ausgegeben. Die Befehle haben die Bezeichnung LDCR (Load CRU) zum Ausgeben und STCR (Store CRU) zum Einlesen.
Je nachdem, wieviele Bits übertragen werden, wird jeweils das höherwertige Byte oder das ganze betroffene Register verwendet. Ein Beispiel:

```
 LI    R12,>1300
 LI    R5,>AA00
       LDCR R5,8
```

Im Register R5 ist damit folgende Bitmaske enthalten:
1010101000000000
Die hinteren 8 Nullen interessieren uns dabei nicht, denn mit LDCR R5,8 geben wir nur 8 Bits aus, es handelt sich also um einen Byte-Befehl. Die Ausführung hat damit folgende Wirkung:
CRU-Adresse >1300 wird 0 gesetzt
CRU-Adresse >1302 wird 1 gesetzt
CRU-Adresse >1304 wird 0 gesetzt
CRU-Adresse >1306 wird 1 gesetzt
CRU-Adresse >1308 wird 0 gesetzt
CRU-Adresse >130A wird 1 gesetzt
CRU-Adresse >130C wird 0 gesetzt
CRU-Adresse >130E wird 1 gesetzt
Die Ausgabe beginnt also mit dem niederwertigsten Bit im Register R5 (Byte) an die niederwertigste CRU-Adresse.
Die Umkehrung dieses Befehls ist STCR. Dabei werden die CRU-Speicheradressen in das Register eingelesen. Die Ausführung ist aber sonst gleich wie bei LDCR.
Beispiele für die Verwendung der CRU-Befehle mit der dazugehörigen Hardware hatten wir mit dem Epromer (TI-REVUE 5 und 6/85) und der 8-Kanal Ein-/Ausgabe (TI-REVUE 2/84) schon vorgestellt, deshalb will ich hier dies nicht nochmal näher erläutern.

Heiner Martin

```
***************************
***************************
**                       **
**                       **
**      CODEWANDLER      **
**      ===========      **
**                       **
**   TMS9900 ROUTINEN    **
**                       **
**                       **
**          BY           **
**                       **
**                       **
**     ARTUR KLENNER     **
**                       **
**                       **
** DUISBURG,DEN 8.10.85  **
**                       **
**                       **
***************************
***************************
```

CODEWANDLER

Das Assemblerprogramm „Codewandler" besteht aus den 3 Teilprogrammen: "HEDE", "HESP" und "DEHE".

HEDE wandelt eine maximal 4stellige Hexadezimale Zahl in eine Dezimalzahl um.

Bei dem Programm HESP wird diese Dezimalzahl noch zusätzlich in einen Dezimalen Adreßwert (–32768 bis 32767) umgewandelt.

Das Programm DEHE wandelt eine dezimale Zahl oder Adreßwert (–32768 bis 65536) in den entsprechenden hexadezimalen Wert um.

Die Programme executieren ca. 4mal schneller als jedes mit bekannte Basic Programm mit gleicher Funktion, und können bei Einbau in z.B. ein Disassembler Programm dessen Ausführungsgeschwindigkeit glatt verdoppeln.

```
*
* TMS 9900 ASSEMBLER  VERSION 1.1  FUER TI EXTENDED BASIC
*
* QUELLPROGRAMMNAME:SCDW
* OBJEKTCODENAME:    OCDW
*
* LADEN MIT CALL LOAD("DSK1.OCDW")
*
* DIESES PROGRAMM ENTHAELT ROUTINEN FUER DIE SCHNELLE
* UMWANDLUNG VON DEZIMALZAHLEN IN HEXADEZIMAL UND UM-
* GEKEHRT,WOBEI DIE HEXADEZIMALE ZAHL MAXIMAL 4 STELLEN
* HABEN DARF.
* FUER DIE UMWANDLUNG VON DEZIMALEN SPEICHERADRESSEN
* IN HEXWERTE KANN DAS PROGRAMM "DEHE" BENUTZT WERDEN.
* NEGATIVE DEZIMALE SPEICHERWERTE WERDEN MIT VORANGESTELLTEM
* MINUSZEICHEN EINGEGEBEN.
* DER AUSGEGEBENE HEXWERT IST IMMER 4 STELLIG.
*
*
* DIE BEFEHLE LAUTEN:
* CALL LINK("HEDE",STRINGVARIABLE,NUM.VARIABLE) =HEX IN DEZIMAL
* CALL LINK("HESP",STRINGVARIABLE,NUM.VARIABLE) =HEXADRESSE IN DEZ.ADRESSWERT
* CALL LINK("DEHE",NUM.VARIABLE,STRINGVARIABLE) =DEZIMAL IN HEX
*
*
BUFF   BSS  8           BUFFER FUER ASCII-ZEICHEN
USRWP  BSS  32          EIGENE ARBEITSREGISTER
STACK  BSS  2           BUFFER FUER RUECKSPRUNGADRESSEN


STATUS EQU  >837C       STATUSREGISTER
GPLWP  EQU  >83E0       ARBEITSREGISTER FUER GPL
STRREF EQU  >2014       UEBERNIMMT STRINGPARAMETER
STRASG EQU  >2010       UEBERGIBT STRINGPARAMETER
NUMREF EQU  >200C       UEBERNIMMT NUM.PARAMETER
NUMASG EQU  >2008       UEBERGIBT NUM.PARAMETER
FAC    EQU  >834A       FPOINT-AKKU
ARG    EQU  >835C       ARGUMENT-AKKU FUER MATHEM.ROUTINEN
XMLLNK EQU  >2018       INT.MASCHINENSPRACHE ROUTINE
```

```
FADD    EQU  >0D80         FPIONT ADDITION
CFI     EQU  >12B8         FPOINT TO INTEGER
CIF     EQU  >20           INTEGER TO FFOINT

B3      EQU  >3000         KONSTANTE >3000
B4      EQU  >4000         KONSTANTE >4000
BYT4    BYTE 4             BYTEKONSTANTE
SUBW    DATA >BDFA,>3724,>0000,>0000  KONSTANTE DEZ.=-65536
ADDW    DATA >4206,>3724,>0000,>0000  KONSTANTE DEZ.=+65536

        DEF  HEDE,HESP,DEHE

        EVEN

*      *****************
*     *******************
*     **               **
*     **  CODEWANDLER  **
*     **  ===========  **
*     **               **
*     *******************
*      *****************
*
*------------------------------------------------------------------------------
*
*          ********
*          *      *
*          * HESP *
*          *      *
*          ********
*
* UMWANDLUNG EINES HEXWERTES IN EINEN DEZIMALEN ADRESSWERT.

HESP    LWPI USRWP
        BL   @STRL         HEXSTRING LESEN - LIEGT DANN ALS ASCII-WERT VOR
        BL   @AHE          UMWANDELN ASCII IN HEXADEZIMAL
        BL   @HDW          UMWANDELN HEXWERT IN DEZIMALWERT
        BL   @AGE          UND AUSGEBEN
        BL   @RET          ZURUECK ZUM BASIC

*       ********
*       *      *
*       * HEDE *
*       *      *
*       ********
*
* UMWANDLUNG EINES HEXWERTES IN EINEN DEZIMALWERT.

HEDE    LWPI USRWP
        BL   @STRL         SIEHE HESP
        BL   @AHE            II   II
        BL   @HDW            II   II
        BL   @PRN          FAC AUF NEGATIVEN WERT PRUEFEN UND KORRIGIEREN
        BL   @AGE          SIEHE HESP
        BL   @RET            II   II

*SUBROUTINE AUF NEGATIVEN WERT PRUEFEN UND KORRIGIEREN

PRN     MOV  @FAC,@FAC     FAC AUF Ø PRUEFEN
```

```
        JEQ  MP1            FAC=Ø?DANN RETURN
        LI   R9,B4          PRUEFWERT LADEN
        C    R9,@FAC        1. WORT IM FAC NEGATIV?
        JGT  KORE           DANN SPRINGE ZUR KORREKTURROUTINE
        JMP  MP1            NICHT NEGATIV - DANN RETURN
KORE    MOV  R11,@STACK     RUECKSPRUNG SICHERN
        LI   R9,ADDW        ZEIGER AUF KONSTANTE SETZEN
        BL   @FPADD         ZUR ADDITION VERZWEIGEN
        MOV  @STACK,R11     ALTE RUECKSPRUNGADRESSE LADEN
MP1     B    *R11           RETURN

* SUBROUTINE STRING LESEN

STRL    LIMI Ø
        CLR  RØ
        LI   R1,1           1.PARAMETER AUS LINK SOLL UEBERNOMMEN WERDEN!
        LI   R2,BUFF        ZEIGER AUF BUFFERANFANG SETZEN
        MOVB @BYT4,*R2      STRING DARF MAXIMAL 4 BYTE LANG SEIN
        BLWP @STRREF        STRING UEBERNEHMEN
        LIMI 2
        B    *R11           RETURN

* SUBROUTINE HEXWERT IN DEZIMALWERT UMWANDELN

HDW     MOV  R7,@FAC        R7 ENTHAELT DEN HEXWERT - AUF FAC LADEN
        LIMI Ø
        BLWP @XMLLNK        MASCHINENSPRACHEROUTINE UMWANDELN
        DATA CIF            HEXWERT IN DEZIMALWERT
        LIMI 2
        B    *R11           RETURN

* SUBROUTINE WERT IN LINK-PARAMETER SCHREIBEN

AGE     CLR  RØ
        INC  R1             IN 2.PARAMETER SOLL GESCHRIEBEN WERDEN
        LIMI Ø
        BLWP @NUMASG        SCHREIBEN
        LIMI 2
        B    *R11           RETURN

* SUBROUTINE ZURUECK ZUM BASIC

RET     CLR  RØ
        MOVB RØ,@STATUS
        LWPI GPLWP
        B    @>ØØ7Ø

*  SRT CODEWANDLER ASCII IN HEX
*  LIEST 4 ASCII ZEICHEN AUS
*  BUFFER UND WANDELT SIE UM IN
*  EINE HEX-ADRESSE

AHE     MOV  R11,@STACK     RUECKSPRUNG SICHERN
        LI   R8,BUFF        ZEIGER AUF ASCII-ZEICHEN SETZEN
        CLR  R5
        CLR  R6
        CLR  R7
        MOVB *R8,R5         ANZAHL ASCII ZEICHEN IN R5
        SWPB R5             KORRIGIEREN
M2      INC  R8             ZEIGER AUF NAECHSTES ZEICHEN SETZEN
```

```
        BL    @UMW              ZEICHEN UMWANDELN
        SOC   R6,R7             UMGEWANDELTES ZEICHEN ZU R7 DAZUGEBEN
        DEC   R5                ALLE ZEICHEN UMGEWANDELT?
        JEQ   EXIT              DANN SPRINGE
        SLA   R7,4              SONST PLATZ MACHEN FUER NAECHSTES ZEICHEN
        JMP   M2                UND UMWANDELN

********************************

UMW     MOVB *R8,R6             ZEICHEN IN R6 LADEN
        ANDI R6,>F000           H-NYBBLE AUSBLENDEN
        LI    R9,B3             MASKE LADEN
        XOR   R9,R6             H-NYBBLE=3?
        JNE   M3                NEIN-DANN SPRINGE
        MOVB *R8,R6             JA-DANN WEITER
        ANDI R6,>0F00           L-NYBBLE AUSBLENDEN
        JMP   M4                L-NYBBLE=HEXWERT DES ASCII-ZEICHENS

M3      MOVB *R8,R6             SIEHE OBEN
        ANDI R6,>F000             II    II
        LI    R9,B4               II    II
        XOR   R9,R6               II    II
        JNE   FEHLER            H-NYBBLE IST NICHT=4...ALSO FEHLER
        MOVB *R8,R6
        ANDI R6,>0F00           L-NYBBLE AUSBLENDEN
        JEQ   FEHLER            =0...ALSO FEHLER
        CI    R6,>0600          ODER >6?
        JGT   FEHLER            AUCH FEHLER
        AI    R6,>0900          L-NYBBLE+9=HEXWERT

M4      SWPB R6                 KORRIGIEREN
        B     *R11              RETURN

FEHLER CLR   R7                 BEI FEHLER 0 AUSGEBEN
EXIT    MOV   @STACK,R11        RETURNADRESSE GEWINNEN
        B     *R11              UND RETURN

*       ********
*       *      *
*       * DEHE *
*       *      *
*       ********
*
* UMWANDLUNG EINES DEZIMALWERTES IN EINEN HEXWERT

DEHE    LWPI USRWP
        BL    @WEL              DEZIMALWERT LESEN
        BL    @ZPR              ZAHL AUF > 32767 UND < 0 PRUEFEN UND KORRIGIEREN
        BL    @DHW              UMWANDLUNG FPIONT IN INTEGER
        BL    @IAS              UMWANDLUNG INTEGER IN ASCII
        BL    @ASTR             AUSGABE
        BL    @RET              ZURUECK ZUM BASIC

* SUBROUTINE ZAHL PRUEFEN UND KORRIGIEREN

ZPR     MOV   @FAC,R9           FAC PRUEFEN
        CI    R9,>BC00          ERST AUF NEGATIVE WERTE
        JH    MJ1               NEGATIV? IST RICHTIG!
        CI    R9,>4203          FAC AUF WERTE > 30000 PRUEFEN
        JH    KORR              GROESSER? DANN SPRINGE ZUR KORREKTURROUTINE
```

```
        JEQ  MJ3             GLEICH? DANN SPRINGE ZUM NAECHSTEN PRUEFVORGANG!
        JMP  MJ1             KLEINER! DANN ALSO ZURUECK ZUM HAUPTPROGRAMM!
MJ3     MOV  @FAC+2,R9       FAC AUF WERTE > 32767 PRUEFEN
        CI   R9,>1B44
        JHE  KORR            GROESSER ODER GLEICH?DANN ZUR KORREKTUR!
        JMP  MJ1             KLEINER! ALSO RETOUR!
KORR    LI   R9,SUBW         ZEIGER AUF KONSTANTE SETZEN
        MOV  R11,@STACK      RUECKSPRUNG SICHERN
        BL   @FPADD          ZUR ADDITION VERZWEIGEN
        MOV  @STACK,R11      ALTE RUECKSPRUNGADRESSE LADEN
MJ1     B    *R11            RETURN

* SUBROUTINE WERT LESEN

WEL     LIMI Ø
        CLR  RØ
        LI   R1,1            1.PARAMETER AUS LINK SOLL UEBERNOMMEN WERDEN
        BLWP @NUMREF         WERT UEBERNEHMEN
        LIMI 2
        B    *R11            RETURN

* SUBROUTINE DEZIMALWERT IN HEXWERT UMWANDELN

DHW     LIMI Ø
        BLWP @XMLLNK         UMWANDELN
        DATA CFI             IN INTEGERWERT
        LIMI 2
        MOV  @FAC,R7         INTEGERWERT IN R7 GEBEN
        B    *R11            RETURN

* SUBROUTINE STRING IN LINK-PARAMETER SCHREIBEN

ASTR    CLR  RØ
        LI   R2,BUFF
        INC  R1              IN 2.LINK-PARAMETER
        LIMI Ø
        BLWP @STRASG         SCHREIBEN
        LIMI 2
        B    *R11            RETOUR

* SUBROUTINE INTEGER IN ASCII UMWANDELN (INTEGER ENTSPRICHT HEXWERT)

IAS     LI   R6,BUFF         ANFANGSADRESSE FUER ASCII-BUFFER LADEN
        MOVB @BYT4,*R6       DARF 4 ZEICHEN LANG SEIN
        INC  R6              NAECHSTES ZEICHEN
        LI   R5,4            ZAEHLER FUER 4 ZEICHEN
MA3     SRC  R7,12           HEXWERT IN POSITION BRINGEN
        MOV  R7,R8
        ANDI R8,>ØØØF        HEXWERT AUSBLENDEN
        CI   R8,>ØØØ9        HEXWERT >9 ?
        JH   MA1             DANN SPRINGE ZUR KORREKTUR!
MA2     AI   R8,>ØØ3Ø        >3Ø DAZU ADDIEREN (ASCII >3Ø BIS >39 (Ø-9))
        SWPB R8              KORRIGIEREN
        MOVB R8,*R6          UND IN BUFFER LADEN
        INC  R6              AUF NAECHSTEN BUFFERPLATZ ZEIGEN
        DEC  R5              ZAEHLER -1  ZAEHLER=Ø?
        JNE  MA3             NEIN - DANN NAECHSTES ZEICHEN!
        B    *R11            RETOUR
MA1     AI   R8,>ØØØ7        >7 ZUM HEXWERT DAZU ADDIEREN (ASCII >41 BIS >46)
        JMP  MA2             UND ZURUECK                      (A  BIS  F)
```

```
* SUBROUTINE FPIONT-ADDITION (FAC+ARG= IN FAC)

FPADD  LI    R5,4           4 WORTE MUESSEN INS ARG GELADEN WERDEN
       LI    R8,ARG         ZEIGER AUF ARG SETZEN
MX1    MOV   *R9+,*R8+      KONSTANTE INS ARG LADEN
       DEC   R5             ALLES GELADEN?
       JNE   MX1            NEIN - NAECHSTES WORT
       LIMI  0
       BLWP  @XMLLNK        ZUR ADDITION
       DATA  FADD
       LIMI  2
       B     *R11           RETURN

       END
```

Ein Beispiel für die Anwendung der Assemblerbefehle ist das anschließend abgedruckte Basicprogramm, welches einen Hexdump von einem Speicherbereich liefert.

Artur Kleuner

```
100 !**********************
110 !*                    *
120 !*   H E X D U M P    *
130 !*                    *
140 !* COPYRIGHT 1985 BY  *
150 !*   ARTUR KLENNER    *
160 !*                    *
170 !* BENOETIGTE GERAETE *
180 !*  TI99/4A KONSOLE   *
190 !*     EXT.BASIC      *
200 !*SPEICHERERWEITERUNG *
210 !*  DISKETTENSTATION  *
220 !*                    *
230 !*  SPEICHERBELEGUNG  *
240 !*     1704 BYTES     *
250 !*                    *
260 !**********************
270 CALL CLEAR
280 PRINT "DIESES PROGRAMM F
ERTIGT      EINEN HEXDUMP AUS
 DEM        SPEICHERBEREICH A
N."
290 PRINT "DER SPEICHERBEREI
CH WIRD     DURCH EINGABE DER
 START-UND ENDADRESSE IN HEX
ADEZIMAL    FESTGELEGT."
300 PRINT "DURCH WAHL DES AU
SGABE-      FORMATES KANN DER
 AUSDRUCK  DEM EINZUGEBENDEN
 GERAET AN-GEPASST WERDEN."
310 PRINT "WENN KEIN GERAET
EINGEGEBEN WURDE,ERFOLGT DER
 AUSDRUCK  AUF DEM BILDSCHIR
M."
320 PRINT "VOR PROGRAMMSTART
 IST        CALL INIT
            CALL LOAD('DSK1.O
CDW')       EINZUGEBEN."
330 PRINT
340 PRINT "DRUECKE EINE TAST
E!"
350 CALL KEY(0,K,S):: IF S=0
 THEN 350
360 DIM W(32),W$(32)
370 CALL CLEAR
380 INPUT "VON,BIS:":A$,B$
390 INPUT "GERAETENAME:":F$
400 INPUT "FORMAT(2-32):":F
:: IF F<2 OR F>32 THEN 400
410 IF F$<>"" THEN DATEI=1 :
: OPEN #DATEI:F$ ELSE DATEI=
0
420 CALL LINK("HEDE",A$,A)
430 CALL LINK("HEDE",B$,B)
440 S=1
450 CALL LINK("DEHE",A,H$)::
 PRINT #DATEI:H$;"    ";
460 K=A :: IF K>32767 THEN K
=-65536+K
470 CALL PEEK(K,W(S),W(S+1))
480 WH=(W(S)*256)+W(S+1):: C
ALL LINK("DEHE",WH,H$)
490 PRINT #DATEI:H$;" ";
500 IF W(S)<32 OR W(S)>126 T
HEN W(S)=42
510 IF W(S+1)<32 OR W(S+1)>1
26 THEN W(S+1)=42
520 A=A+2 :: S=S+2 :: IF F>S
 THEN 460
530 S=S-1
540 PRINT #DATEI:"   ";
550 FOR I=1 TO S
560 PRINT #DATEI:CHR$(W(I));
570 NEXT I
580 PRINT #DATEI:CHR$(13)!WA
GENRUECKLAUF
590 IF B>A THEN 440
600 GOTO 380
```

MULTIPLAN FÜR DENTI-TEIL II

Bei dem hier vorgestellten Arbeitsblatt (Modell) handelt es sich um eine Tabelle, die aus 6 einzugebenden Werten (Parametern) eine Tabelle bestehend aus 116 Werten in ca. 50 Sekunden errechnet.

Der Sachverhalt sei hier nur kurz skizziert. Wer mehr Informationen über den fotografischen Hintergrund erfahren möchte, findet alles dazu wichtige in: Tölke, A. u. i. Tölke, „Fotografie und Kinematografie (TI-58C/TI-59) – 19 Programme für Amateur und Berufsfoografen“ Vieweg Verlag Wiesbaden 1984.

Bei geblitzten Nahaufnahmen ist es zweckmäßig, eine Blitzlampe direkt vorne am Objektiv zu befestigen (z.B. einen Ringblitz) und die zweite Blitzlampe als Effektlicht in seitlicher Gegenlichtposition einzusetzen. Dabei interessiert in Abhängigkeit vom eingestellten Abbildungsmaßstab der dann zu wählende Blendenwert und der Abstand der zweiten Blitzlampe. Die sechs Ausgangswerte (Eingabeparameter) im Kopf der Tabelle befinden sich im Feldbereich R1:6C4. Damit ist eine Anpassung der Wertetabelle an jede Kameraausrüstung innerhalb kürzester Zeit (ca. 50 Sekunden) möglich. Aus der errechneten und gedruckten Tabelle kann der jeweilige Blendenwert und der Abstand der Effektlichtlampe (in cm) für verschiedene Kontrastverhältnisse direkt abgelesen werden. Die Leitzahl der beiden Blitzlampen sind die Werte, die der Hersteller in der Bedienungsanleitung für 21 DIN angibt. Die Umrechnung auf die Lichtempfindlichkeit des verwendeten Filmmaterials erfolgt automatisch und wird im Feldbereich R8:9C4 zur Kontrolle angegeben. Der Korrekturwert (KW) ist der Abstand zwischen der vorderen Reflektorebene der im Objektiv angebrachten Blitzlampe (Vorderlicht) und der Filmebene. Dabei muß das Objektiv auf die Entfernung Unendlich eingestellt

```
       1         2          3          4         5             6          7
 1 Leitzahl Vorderlicht (LV):                *fuer DIN:        21        *
 2 Leitzahl Effektlicht (LE):                *fuer DIN:        21        *
 3      Brennweite in mm   (f):              *1. Blitz:                  *
 4         Pupillenmasstab (P):              *2. Blitz:                  *
 5  Korrekturwert in mm (KW):                *Objektiv:                  *
 6  aktueller DIN-Wert (DIN):                *  Kamera:                  *
 7 ***************************************************************************
 8 Leitzahl Vorderlicht(LVa):    0.00 fuer DIN :       0
 9 Leitzahl Effektlicht(LEa):    0.00 fuer DIN :       0
10 ----------------------------------------------------------------------------
11                                             Abstand des Effeklichts
12 Auszug Masstab  Entfernung  akt.     bei einem Kontrastverhaeltnis
13 in mm als Betrag Lampe 1   Blende      1:1          1:2          1:4
14  (z')   (/B/)       (Em)     (k)              Entfernung in cm
15 ----------------------------------------------------------------------------
16          #DIV/0!    #DIV/0!   #VALUE!  #VALUE!     #VALUE!     #VALUE!
17   -10    #DIV/0!    #DIV/0!   #VALUE!  #VALUE!     #VALUE!     #VALUE!
18   -20    #DIV/0!    #DIV/0!   #VALUE!  #VALUE!     #VALUE!     #VALUE!
19   -30    #DIV/0!    #DIV/0!   #VALUE!  #VALUE!     #VALUE!     #VALUE!
20   -40    #DIV/0!    #DIV/0!   #VALUE!  #VALUE!     #VALUE!     #VALUE!
21   -50    #DIV/0!    #DIV/0!   #VALUE!  #VALUE!     #VALUE!     #VALUE!
22   -60    #DIV/0!    #DIV/0!   #VALUE!  #VALUE!     #VALUE!     #VALUE!
23   -70    #DIV/0!    #DIV/0!   #VALUE!  #VALUE!     #VALUE!     #VALUE!
24   -80    #DIV/0!    #DIV/0!   #VALUE!  #VALUE!     #VALUE!     #VALUE!
25   -90    #DIV/0!    #DIV/0!   #VALUE!  #VALUE!     #VALUE!     #VALUE!
26  -100    #DIV/0!    #DIV/0!   #VALUE!  #VALUE!     #VALUE!     #VALUE!
27  -110    #DIV/0!    #DIV/0!   #VALUE!  #VALUE!     #VALUE!     #VALUE!
28  -120    #DIV/0!    #DIV/0!   #VALUE!  #VALUE!     #VALUE!     #VALUE!
29  -130    #DIV/0!    #DIV/0!   #VALUE!  #VALUE!     #VALUE!     #VALUE!
30  -140    #DIV/0!    #DIV/0!   #VALUE!  #VALUE!     #VALUE!     #VALUE!
31  -150    #DIV/0!    #DIV/0!   #VALUE!  #VALUE!     #VALUE!     #VALUE!
32  -160    #DIV/0!    #DIV/0!   #VALUE!  #VALUE!     #VALUE!     #VALUE!
33  -170    #DIV/0!    #DIV/0!   #VALUE!  #VALUE!     #VALUE!     #VALUE!
34 ----------------------------------------------------------------------------
```

sein und ein Zwischenring bzw. Balgengerät darf sich nicht zwischen Kameragehäuse und Objektiv befinden.

Ablesebeispiel: Unter den im Kopf angegebenen Bedingungen wird für eine Aufnahme eine Auszugsverlängerung von 130 mm (z') gewählt. Die Blende beträgt dann laut Multiplan-Modell 22 und bei einem gewünschten Kontrastverhältnis von 1:2 (entspricht leichtem Gegenlicht) muß die zweite Blitzlampe 50 cm vom Objekt entfernt sein. Die Aufnahme gelingt jetzt hundertprozentig.

Schwerpunkt der weiteren Betrachtung ist aber nicht die Fotografie, sondern die Umsetzung der für die Berechnung notwendigen mathematischen Formeln in Multiplan-Formeln.

Beim Aufbau eines derartigen Multiplan-Modells sollte mit der Definierung der Spaltenbreite begonnen werden. Hierbei wurde folgende Zeichenbreite gewählt:

Spalte 1: Zeichenanzahl beträgt	6
Spalte 2: Zeichenanzahl beträgt	10
Spalte 3: Zeichenanzahl beträgt	10
Spalte 4: Zeichenanzahl beträgt	8 (Standard)
Spalte 5: Zeichenanzahl beträgt	10
Spalte 6: Zeichenanzahl beträgt	12
Spalte 7: Zeichenanzahl beträgt	9

Durch diese Formatierungsbreite ist gewährleistet, daß die Spalte für die Eingabeparameter ohne Betätigen der Pfeiltasten sich auf dem Bildschirm befindet und links davon die Erläuterung. Die Parameterfelder erhalten die Namen, die links vor dem Doppelpunkt in Klammern stehen. Ebenfalls wird der Text entsprechend dem Vorbild eingegeben.

Für die Berechnung der aktuellen Leitzahl (LVa und LEa) gelten folgende Zusammenhänge:
wenn der aktuelle DIN-Wert größer 21 ist gilt (1a)
wenn der aktuelle DIN-Wert kleiner 21 ist gilt (1b)

$$LVa = LV*6.\ \text{Wurzel aus}\ 2^{(\ 21\text{-}DIN)} \qquad (1a)$$

$$LVa = LV:6.\ \text{Wurzel aus}\ 2^{(21\text{-}DIN)} \qquad 1b)$$

Die senkrechten Striche vor und hinter dem Ausdruck 21-DIN bedeuten: „bilde den Betrag", also jetzt positiver Wert.

Sinngemäß das Gleiche wie für LVa gilt für die Berechnung von LEa, nur das hier natürlich von LE (Leitzahl Effektlicht bei 21 DIN) ausgegangen werden muß. Da Multiplan nur Quadratwurzeln berechnen kann, wird entsprechend folgender Regel berechnet: Die n'te Wurzel aus einer Potenz wird gezogen, indem man den Exponenten durch n teilt. Dadurch nimmt (1a) und (1b) folgende Gestalt an:

$$LVa = LV*2^{(\ 21\text{-}DIN\ ):6} \qquad (1a')$$

$$LVa = LV:2\ ^{(21\text{-}DIN):6} \qquad (1b')$$

Da je nach gewähltem DIN-Wert eine leicht geänderte Formel zur Berechnung benutzt werden muß (1a oder 1b bzw. 1a' oder 1b'), bietet sich die Multiplan-IF-Anweisung direkt an. Diese ist aus der Tabelle zu entnehmen. Für die Berechnung von LVa gilt M-Formel 1 und für LEa gilt M-Formel 2.

Innerhalb dieser Tabelle wurden oft mehr Klammern gesetzt als eigentlich notwendig sind. Dies geschah aus Gründen der besseren Übersicht, denn in der IF-Anweisung sind nach der Vergleichsoperation (hier DIN größer 21) die Formel 1a und 1b, jeweils durch Komma getrennt, der schon besprochene Wert1 und Wert2. Diese beiden Anweisungen (eigentlich eine komplexe Formel) werden unter dem Kommando „Value" in das Feld R8C4 (M-Formel 1) und R9C4 (M-Formel 2) eingetragen. Davor oder danach erhalten diese beiden Felder den Namen LVa bzw. LEa. Die Felder R8C6 und R9C6 sollten mit dem Wert (!) DIN belegt werden, damit bei jeder veränderten Eingabe des Felds mit dem Namen DIN auch diese Felder der aktuellen Wert übernehmen. Natürlich wäre dies auch durch die Eingabe der relativen Adresse R[9–2]C[–2] zu erreichen.

Das Feld R16C1 erhält den Wert 180, denn dies ist schon ein Extremwert, der aber für viele Situationen einen Abbildungsmaßstab berechnen läßt, der oberhalb 1:1 (größer 1,0) liegt. Unterhalb dieses Feldes ist die M-Formel 3 einzugeben und 16 mal nach unten zu kopieren. Damit wird erreicht, daß jedes Feld einen Wert erhält, der um 10 geringer ist als der des jeweils direkt darüber liegenden Feldes. Achtung (!) Nach ‚C' darf keine eckige Klammer folgen, da der Bezugspunkt der Adresse ja innerhalb der gleichen Spalte liegt, also C ohne Felderanzahl. Die Angabe minus 10 ist eine arithmetische Operation und keine Adressenangabe.

In der nächsten Spalte muß nun der Abbildungsmaßstab als Betrag (B) für jede Zeile der Tabelle berechnet werden, da andere Formeln darauf Bezug nehmen. Hierfür gilt mathematisch (2)

$$B = Z':f$$

Für Multiplan gilt die M-Formel 4. Sie wird im Feld R16C2 eingetragen und 17 (!) mal nach unten kopiert. Der Wert für f (Brennweite) wird vom Feld mit dem Namen f erreicht und z' durch die relative Adresse.

Da für weitere Berechnungen auch der Abstand zwischen der vorne am Objektiv angebrachten Blitzlampe (Vorderlicht) bekannt sein muß, erfolgt die Berechnung in Meter in Spalte 3. Hierfür gilt die Beziehung (3)

$$Em = \frac{f\,\frac{(B+1)^2}{B} - z' - KW}{1000} \qquad (3)$$

Für den Eintrag in Feld R16C3 gilt M-Formel 5. Auch diese Formel wird 17 mal nach unten kopiert. KW und f sind Namen für schon erwähnte Felder (Parameterfelder).

Nun wird der aktuelle Blendenwert bezogen auf das Vorderlicht berechnet. Dabei gilt (4).

$$k = \frac{LVa}{Em\frac{(B+1)}{p}} \qquad (4)$$

Es handelt sich hierbei um die umgewandelte und in der Fotografie bekannte Leitzahlformel. Falls der Pupillenmaßstab (P) nicht bekannt ist (Berechnung siehe angegebene Literaturquelle) muß der Wert 1 dafür eingesetzt werden. Für Multiplan gilt M-Formel 6. Ihre Eintragung geschieht im Feld R16C4 und auch sie muß 17 mal nach unten kopiert werden.

Für die Berechnung der Entfernung der zweiten Blitzlampe muß in jedem Fall von der im Feld LEa berechneten und dort stehenden ak-

tuellen Leitzahl ausgegangen werden. Dabei gilt (5)

$$Ecm = \frac{LEa*100}{k(\frac{B}{p}+1)} \qquad (5)$$

Bei einem Kontrastverhältnis von 1:1 (Vorderlicht ist genauso stark wie das Gegenlicht) gilt der Blendenwert (k), so wie er in der Spalte 4 steht (M-Formel 7). Für einen Beleuchtungskontrast von 1:2 muß k mit dem Wert „Wurzel aus 2" multipliziert werden. Dadurch entsteht eine Blendenzahl, die um eine Blendenwertstufe niedriger ist als die für das Vorderlicht berechnete. Hierfür muß somit M-Formel 8 benutzt werden. Diese veränderte Situation ist aus dem Teil-Term *SQRT(2) erkennbar. Bei einem Kontrastverhältnis 1:4 muß der Blendenwert mit 2 multipliziert werden. Deshalb ist in dieser Spalte die M-Formel 9 zu benutzen. In allen drei Fällen müssen die jeweiligen Formeln wiederum 17 mal nach unten kopiert werden.

Gibt man jetzt die Werte in die 6 Parameterfelder ein und drückt danach die Taste RECLAC (FCTN 8), erscheinen innerhalb von 50 Sekunden alle gewünschten Werte dieser Tabelle. Das nun fertige Arbeitsblatt kann entweder auf Diskette abgespeichert oder als Tabelle ausgedruckt werden. In kürzester Zeit stehen auf diese Weise nun Tabellen zur Verfügung, die in unserem gewählten Beispiel die verschiedensten Situationen – bezogen auf die eigene fotografische Ausrüstung – berücksichtigen.

Schlußbemerkung

Es ist erstaunlich, in welch' kurzer Zeit es möglich ist, ein derartiges Multiplan-Modell aufzustellen. Vorausgesetzt, der Benutzer beherrscht die Kommandos und das Umsetzen einer mathematischen Formel in eine Multiplan-Formel. Zum Letztgenannten gehört ein wenig Übung, weil Multiplan kein AOS-Rechner ist, sondern die Rechnung in der Formel vorne beginnend schrittweise nach hinten gehend ausführt. Deshalb muß mit relativ viel Klammern gearbeitet werden. Dabei ist lediglich zu beachten, daß eine Formel nicht mehr als 85 Zeichen enthält. Bei syntaktischen Fehlern reagiert Multiplan so, daß es die Formel nicht annimmt und meistens den Fehler durch den Menücursor sogar anzeigt. Ein Grund mehr, sich intensiver mit Multiplan zu beschäftigen, denn die Grenze des Möglichen zu finden, ist auch in Bezug auf Stringverarbeitung nicht ganz einfach, weil vieles noch gar nicht ausgelotet wurde.

Ov. A. Tölke

LOAD INTERRUPT

„LOAD-INTERRUPT? Was ist denn das?", werden einige von Ihnen sicher denken. Daß nur wenige über diese Möglichkeit des TI Bescheid wissen, liegt wieder einmal an der zurückhaltenden Informationspolitik des Herstellers. Dabei kann der LOAD-INTERRUPT sehr nützlich sein. Doch zuerst soll erklärt werden, was dieser Interrupt bewirkt und wie man ihn anspricht.

Bei einem LOAD-INTERRUPT werden der Workspace Pointer und der Program Counter mit den Adressen, die von)FFFC (dez. –4) aufwärts stehen, geladen, das heißt, daß der Prozessor einen neuen Arbeitsspeicher bekommt und einen Sprung zu einer neuen Adresse ausführt. Dazu kommt noch, daß dieser Interrupt die höchste Priorität nach dem RESET hat, das bedeutet, daß er softwaremäßig nicht unterbunden werden kann.

Soweit zur Theorie, nun folgt die Praxis: Man benötigt einen TI mit Speichererweiterung ()FFFC liegt in ihr) und entweder das Ex-BASIC-, Minimem- oder E/A-Modul (zum Poken). Dann lötet man am I/0-Port (VORSICHT !!! Statische Aufladungen können zu Schäden am TI führen) an die Pins 13 und 21 einen Taster (Pin 13 ist bei Draufsicht auf den Bus der 7te links unten, Pin 21 der 11te links unten). Jetzt kann man in BASIC z.B. einen Reset mit dem Taster ausführen, nachdem man folgendes eingegeben hat:

```
CALL PEEK (0,A,B,C,D)
CALL LOAD (–4,A,B,C,D)
```

Drückt man nun den Taster, wird ein Reset ausgeführt. Diese Funktion bleibt erhalten, bis man andere Werte auf)FFFC legt. Man kann auch mit dem Taster den Debugger starten; wenn man ihn lädt und normal startet, legt dieser von selbst seinen Workspace Pointer und seine Einsprungadresse auf)FFFC. Auf diese Weise kann man z.B. ROM-Module Auslesen und vieles mehr.

Allgemein braucht man, wenn man den Taster zum Starten eines eigenen Maschinenprogramms verwenden will, nur dessen Workspace-Pointer auf)FFFC und seine Einsprungadresse auf)FFFE legen. Zur Rückkehr aus dem Programm muß allerdings der TRWP-Befehl verwendet werden.

Günther Rehm

SPORTTABELLEN

Wenige Sekunden nach der Eingabe 'RUN' startet das Programm mit dem Hauptmenü und den Wahlmöglichkeiten 1 – 8:

1. Aufstellen der Tabelle:

Sind keine Daten vorhanden, weist das Programm Sie darauf hin, wählen Sie dann 3 oder 5.
Sonst sortiert der Computer die Mannschaft nach Punkten oder Toren bei Punktgleichheit.
Das Programm kehrt von selbst zum Hauptmenü zurück.

2. Abbilden der Tabelle:

Sind keine Daten vorhanden, trifft das gleiche wie bei 1 zu.
Sonst zeigt das Programm die Tabelle mit Platzzahl, Mannschaftsname, Tordifferenz und Plus-/Minuspunkte.
Rückkehr zum Hauptmenü durch Drücken einer beliebigen Taste.

3. Neue Anfangswerte und Fehler:

Untermenü mit 4 Wahlmöglichkeiten:
1. Neue Eingabe
– Liganame (28 Zeichen/ z.B.: 1. Fußball Bundesliga)
– Anzahl der Mannschaften:
  – Name (14 Zeichen)
  – Tore/Gegentore
  – gewonnene, unentschiedene, verlorene Spiele
2. Neue Mannschaften (siehe 3.1., nur werden keine Tore/Gegentore und Spiele eingegeben)
3. Fehler verbessern
– Liganame erscheint, Sie können verbessern
– Danach erscheint jede Mannschaft einzeln mit Toren/Gegentoren, Punkten/Minuspunkten. Erkennen Sie einen Fehler, beantworten Sie die Frage "FEHLER?" mit 'J' für ja. Dann werden die Daten wie bei 3.1. aufgelistet und Sie können verbessern. Sind Sie die Werte einer Mannschaft durchgegangen, stellt der Computer wieder die Frage nach einem Fehler. Bei 'N' für nein erscheint die nächste Mannschaft.
4. Rückkehr zum Haupt menü

4. Spieleingabe:

Es werden immer bis zu zehn Mannschaften auf dem Bildschirm aufgelistet mit den Nummern 0–9. Mit den Tasten 'S' und 'D' können Sie blättern. Bei der Wahl einer Mannschaft schlagen Sie bei mehr als zehn Mannschaften die entsprechende "Seite" auf und wählen die Nummer der Mannschaft. Steht die Paarung fest, geben Sie das Ergebnis ein. Jede Mannschaft kann pro Durchgang nur einmal gewählt werden, da sie an einem Spieltag auch nicht öfter spielen kann. Nach der Rückkehr zum Hauptmenü und erneuter Wahl 4 kann wieder jede Mannschaft gewählt werden.
Zum Hauptmenü kehrt man durch 'FCTN 4' (Clear) zurück, das kann auch geschehen, wenn bereits eine Mannschaft gewählt worden ist.

5. u. 6. Laden, bzw. Speichern von Daten (auf Kassette):

Bevor geladen, bzw. gespeichert wird, vergewissert sich der Computer noch der richtigen Wahl, sonst kehrt man zum Hauptmenü zurück.
Nach dem Vorgang des Ladens, bzw. des Speicherns kehrt das Programm sofort zum Hauptmenü zurück.

7. Statistik: Tore

Dieser Programmteil errechnet Ihnen
– Gesamtanzahl der Tore
– Durchschnitt pro Mannschaft
– Durchschnitt pro Spiel

Ferner zeigt es Ihnen für jede Mannschaft
– Gesamtzahl Spiele
– Tore/Gegentore.

Rückkehr zum Hauptmenü durch Drücken einer beliebigen Taste.

8. Ende:

Solange auf dem Bildschirm "ENDE" steht, können Sie durch Drücken einer beliebigen Taste zum Hauptmenü zurückkehren. Erscheint aber "READY", ist das Programm endgültig beendet.

Harald Winkler

GARTEN DER PILZE

Du bist von einem grausamen Herrscher gefangengenommen und mußt in einem von einem Gespenst bewachten Garten Pilze sammeln. Jedesmal, wenn du einen Pilz sammelst, verlierst du Pilzsporen, die zu riesigen Giftpilzen werden, wenn du noch einmal zu dieser Stelle kommst. Diese Giftpilze und die überall im Garten wachsenden Büsche versperren den Weg. Das Gespenst kann allerdings manchmal ein Hindernis überspringen.
Hast du genügend Pilze gesammelt, je nach Stufe zwischen 10 und 50 Stück, wird die Absperrung am linken Bildschirmrand entfernt, und du kannst dorthin flüchten und die Pilze abliefern.
Der böse Herrscher bekommt aber so schnell nicht genug, und schickt dich noch einmal los.
Tip: Dem Gespenst entkommst du am einfachsten, wenn du diagonal gehst.

Anmerkung der Redaktion: Diskettenbesitzer müssen alle REM-Zeilen löschen und vor dem Laden des Programms CALL FILES(1) NEW eingeben.

LERNPARTNER

Die Simplex-Methode ist ein Verfahren der linearen Programmierung und dient zur Lösung von Optimierungsproblemen. Solche treten etwa in der Betriebswirtschaft/Kostenrechnung in Form von Produktions- oder Absatzplanungsaufgaben auf. Die Problemstellung in der Optimierung eines Produktionsprogramms (siehe Beispiel) ergibt sich aus den unterschiedlichen Kapazitäten der einzelnen Maschinen bzw. Fertigungsstellen und aus deren unterschiedlichen Inanspruchnahme durch die einzelnen Werkstücke. Überdies können diese einen unterschiedlichen Beitrag zur Deckung der Fixkosten (Deckungsbeitrag =Nettoerlös minus variable Kosten) oder zum Gewinn beitragen.
Die Frage ist daher, wieviel Stück und welche Produkte erzeugt werden sollen, um ein optimales Betriebsergebnis erwirtschaften zu können.

2. Kommentar zum Beispiel im Anhang:

Bei diesem Beispiel ist das deckungsbeitragsmaximale Produktionsprogramm gefragt. Sieht man sich die erste Zeile des

Bitte lesen Sie weiter auf Seite 36

LISTINGS

```
10 ! *********************
11 ! *                   *
12 ! *   SPORT-TABELLEN  *
13 ! *                   *
14 ! *    Copyright by   *
15 ! *                   *
16 ! *   Harald Winkler  *
17 ! *                   *
19 ! * Benoetigte Geraete *
20 ! *  TI99/4A  Konsole *
21 ! *     Ext. Basic    *
22 ! *    Cassettenrec.  *
23 ! *                   *
26 ! *  Speicherbelegung *
27 ! *     9488 Bytes    *
28 ! *                   *
29 ! *********************
100 ON BREAK NEXT
110 CALL CHAR(95,""):: CALL
CHAR(99,"3C4299A1A199423C")
120 DIM TORE(30),T(30),G(30)
,TG(30),BILANZ(30),GE(30),UN
(30),VERL(30),SP(30),P(30),M
(30),N$(30)
130 REM * HAUPTMENUE
140 CALL CHAR(32,"")
150 DISPLAY AT(4,1)ERASE ALL
:"SPORT-TABELLEN-ERSTELLUNG"
 :: GOSUB 2260 :: DISPLAY AT
(7,1):"c 1985 BY HARALD WINK
LER"
160 DISPLAY AT(9,1):"1 AUFST
ELLEN DER TABELLE":"2 ABBILD
EN DER TABELLE"
170 DISPLAY AT(11,1):"3 NEUE
 ANFANGSWERTE UND FEH-..LER"
:"4 VERAENDERN DER WERTE":"5
 LADEN VON WERTEN"
180 DISPLAY AT(15,1):"6 SPEI
CHERN VON WERTEN":"7 STATIST
IK":"":"8 ENDE"
190 CALL KEY(0,K,S):: IF K<4
9 OR K>56 THEN 190 :: ON K-4
8 GOTO 210,380,550,1250,1650
,1780,1980,2180
200 REM * AUFSTELLEN TABELLE
210 IF LIGA$="" THEN DISPLAY
 AT(24,7):"KEINE DATEN !!!!"
 :: GOTO 170
220 FOR I=1 TO N
230 GOSUB 2450
240 NEXT I
250 DISPLAY AT(4,1)ERASE ALL
:"ERSTELLEN DER TABELLE" ::
GOSUB 2260 :: F=0
260 FOR I=1 TO N-1
270 IF P(I)=P(I+1)THEN 290
280 IF P(I)<P(I+1)THEN 340 E
LSE 350
290 IF M(I)>M(I+1)THEN 340
300 IF M(I)<M(I+1)THEN 350
310 IF TG(I)<TG(I+1)THEN 340
320 IF TG(I)>TG(I+1)THEN 350
330 IF T(I)<T(I+1)THEN 340 E
LSE 350
340 GOSUB 2280
350 NEXT I
360 IF F=1 THEN 250 ELSE 140
370 REM * ABBILDEN TABELLE
380 IF LIGA$="" THEN DISPLAY
 AT(24,7):"KEINE DATEN !!!!"
 :: GOTO 190
390 DISPLAY AT(1,1)ERASE ALL
:LIGA$:"===================
========PL. VEREINSNAME   TO
RE  PKT.--------------------
--------"
400 CALL CHAR(97,"000008083E
08080000000000003E")
410 J=0 :: FOR I=1 TO N :: J
=J+1
420 IF J<20 THEN 440
430 J=19 :: PRINT
440 GOSUB 2450
450 IF TG(I)=0 THEN 470
460 IF TG(I)>0 THEN 480 ELSE
 490
470 DISPLAY AT(4+J,1):USING
"##. #############   0 ##:#
#":I,N$(I),P(I),M(I):: GOTO
510
480 DISPLAY AT(4+J,1):USING
"##. ############# a## ##:#
#":I,N$(I),TG(I),P(I),M(I)::
 GOTO 510
490 TG(I)=-TG(I)
500 DISPLAY AT(4+J,1):USING
"##. ############# b## ##:#
#":I,N$(I),TG(I),P(I),M(I)::
 TG(I)=-TG(I)
510 NEXT I
520 CALL KEY(0,K,S)
530 IF S=0 THEN 520 :: GOTO
140
540 REM * UNTERMENUE
550 CALL CHAR(32,"")
560 DISPLAY AT(4,1)ERASE ALL
:"NEUE ANFANGSWERTE UND FEHL
ERBERICHTIGEN" :: GOSUB 2260
570 DISPLAY AT(8,1):"1 NEUE
EINGABE":"2 NEUE MANNSCHAFTE
N":"3 FEHLER BERICHTIGEN":"4
 MENUE"
580 CALL KEY(0,K,S):: IF K<4
9 OR K>52 THEN 580 :: ON K-4
8 GOTO 600,860,1000,140
590 REM * NEUE EINGABE
600 GOSUB 2240 :: DISPLAY AT
(1,1)SIZE(19):"EINGABE_NEUER
_DATEN" :: CALL HCHAR(2,1,61
,32)
610 ACCEPT AT(3,1):Y$
```

```
620 IF Y$="" THEN 550 ELSE L
IGA$=Y$
630 IF N>0 THEN GOSUB 2380
640 DISPLAY AT(4,1)SIZE(26):
"WIEVIELE_MANNSCHAFTEN_?_ 0"
650 ACCEPT AT(4,25)VALIDATE(
DIGIT)SIZE(-2):Y$ :: IF Y$="
" THEN 650 ELSE N=VAL(Y$)::
IF N<3 OR N>30 THEN 650
660 FOR I=1 TO N
670 DISPLAY AT(6,1)SIZE(5):"
NAME:"
680 DISPLAY AT(8,1)SIZE(15):
"TORE:___...:"
690 DISPLAY AT(11,1)SIZE(15)
:"SPIELSTATISTIK:" :: DISPLA
Y AT(12,1):"GEW.:_.._UN.:_..
_VERL.:_.. __"
700 IF I<10 THEN I$="_"&STR$
(I)&"."
710 DISPLAY AT(24,1)SIZE(14)
:USING "###_MANNSCHAFT":I$
720 ACCEPT AT(7,1)SIZE(14):N
$(I):: IF N$(I)="" THEN 720
730 IF LEN(N$(I))<14 THEN N$
(I)=N$(I)&CHR$(32):: GOTO 73
0
740 ACCEPT AT(8,9)VALIDATE(D
IGIT)SIZE(3):Y$ :: IF Y$=""
THEN 740 ELSE T(I)=VAL(Y$)
750 ACCEPT AT(8,13)VALIDATE(
DIGIT)SIZE(3):Y$ :: IF Y$=""
 THEN 750 ELSE G(I)=VAL(Y$)
760 ACCEPT AT(12,7)VALIDATE(
DIGIT)SIZE(2):Y$ :: IF Y$=""
 THEN 760 ELSE GE(I)=VAL(Y$)
770 ACCEPT AT(12,15)VALIDATE
(DIGIT)SIZE(2):Y$ :: IF Y$="
" THEN 770 ELSE UN(I)=VAL(Y$
)
780 ACCEPT AT(12,25)VALIDATE
(DIGIT)SIZE(2):Y$ :: IF Y$="
" THEN 780 ELSE VERL(I)=VAL(
Y$)
790 P(I)=GE(I)*2+UN(I):: M(I
)=UN(I)+VERL(I)*2
800 DISPLAY AT(15,1):"NAME__
__________TORE___PKT._"
810 DISPLAY AT(16,1):USING "
##############_###:###_##:##
":N$(I),T(I),G(I),P(I),M(I)
820 CALL KEY(0,K,S):: IF S=0
 THEN 820
830 CALL HCHAR(15,1,95,68)
840 GOSUB 2430 :: NEXT I ::
GOTO 550
850 REM * NEUE MANNSCHAFTEN
860 GOSUB 2240 :: DISPLAY AT
(1,1)SIZE(17):"NEUE_MANNSCHA
FTEN" :: CALL HCHAR(2,1,61,3
2)
870 ACCEPT AT(3,1):Y$
880 IF Y$="" THEN 550 ELSE L
IGA$=Y$
890 IF N>0 THEN GOSUB 2380
900 DISPLAY AT(4,1):"WIEVIEL
E_MANNSCHAFTEN_?_ 0__"
910 ACCEPT AT(4,25)VALIDATE(
DIGIT)SIZE(-2):Y$ :: IF Y$="
" THEN 910 ELSE N=VAL(Y$)::
IF N<3 OR N>30 THEN 910
920 J=0 :: FOR I=1 TO N :: J
=J+1
930 IF J<19 THEN 950
940 J=18 :: PRINT :: CALL HC
HAR(24,1,95,32):: CALL HCHAR
(23,23,95,8)
950 DISPLAY AT(5+J,1)SIZE(6)
:"NAME:_"
960 ACCEPT AT(5+J,7)SIZE(14)
:N$(I):: IF N$(I)="" THEN 96
0
970 IF LEN(N$(I))<14 THEN N$
(I)=N$(I)&CHR$(32):: GOTO 97
0
980 NEXT I :: GOTO 550
990 REM * FEHLER BERICHTIGEN
1000 IF LIGA$="" THEN DISPLA
Y AT(24,7):"KEINE DATEN !!!!
" :: GOTO 580
1010 GOSUB 2240 :: DISPLAY A
T(1,1)SIZE(18):"FEHLER_BERIC
HTIGEN" :: CALL HCHAR(2,1,61
,32)
1020 DISPLAY AT(3,1):LIGA$
1030 ACCEPT AT(3,1)SIZE(-28)
:Y$ :: IF Y$="" THEN 1030 EL
SE LIGA$=Y$
1040 FOR I=1 TO N
1050 GOSUB 2450
1060 I$=STR$(I)&"." :: IF I<
10 THEN I$="_"&I$
1070 DISPLAY AT(6,1):USING "
##############_###:###_##:##
":N$(I),T(I),G(I),P(I),M(I):
: DISPLAY AT(24,1)SIZE(14):U
SING "###_MANNSCHAFT":I$
1080 DISPLAY AT(8,1)SIZE(8):
"FEHLER_?"
1090 CALL KEY(0,K,S):: IF K=
78 THEN 1220
1100 IF K=74 THEN 1110 ELSE
1090
1110 DISPLAY AT(10,1)SIZE(5)
:"NAME:" :: DISPLAY AT(11,1)
SIZE(14):N$(I)
1120 DISPLAY AT(13,1)SIZE(14
):USING "TORE:__###:###":T(I
),G(I)
1130 DISPLAY AT(15,1)SIZE(15
):"SPIELSTATISTIK:" :: DISPL
AY AT(16,1):USING "GEW.:_##_
```

LISTINGS

```
UN.:_##_VERL.:_##__":GE(I),U
N(I),VERL(I)
1140 ACCEPT AT(11,1)SIZE(-14
):N$(I):: IF N$(I)="" THEN 1
140
1150 ACCEPT AT(13,8)VALIDATE
(DIGIT)SIZE(-3):Y$ :: IF Y$=
"" THEN 1150 ELSE T(I)=VAL(Y
$)
1160 ACCEPT AT(13,12)VALIDAT
E(DIGIT)SIZE(-3):Y$ :: IF Y$
="" THEN 1160 ELSE G(I)=VAL(
Y$)
1170 ACCEPT AT(16,7)VALIDATE
(DIGIT)SIZE(-2):Y$ :: IF Y$=
"" THEN 1170 ELSE GE(I)=VAL(
Y$)
1180 ACCEPT AT(16,15)VALIDAT
E(DIGIT)SIZE(-2):Y$ :: IF Y$
="" THEN 1180 ELSE UN(I)=VAL
(Y$)
1190 ACCEPT AT(16,25)VALIDAT
E(DIGIT)SIZE(-2):Y$ :: IF Y$
="" THEN 1190 ELSE VERL(I)=V
AL(Y$)
1200 GOSUB 2430
1210 CALL KEY(0,K,S):: IF K=
74 THEN 1050 :: IF K<>78 THE
N 1210
1220 CALL HCHAR(9,1,95,258)
1230 NEXT I :: GOTO 550
1240 REM * SPIELEINGABE
1250 IF LIGA$="" THEN DISPLA
Y AT(24,7):"KEINE DATEN !!!!
" :: GOTO 190
1260 DISPLAY AT(1,1)ERASE AL
L:"SPIELEINGABE":LIGA$ :: CA
LL HCHAR(3,1,61,32):: DISPLA
Y AT(24,1):"CLEAR MENUE" ::
Z=1 :: I1,I2=0
1270 DISPLAY AT(11,1):"SEITE
";Z
1280 CALL HCHAR(12,1,32,320)
:: FOR I=0 TO 9 :: IF I+1+(Z
-1)*10>N THEN 1290 :: DISPLA
Y AT(I+12,1):I;" ";N$(I+1+(Z
-1)*10)
1290 NEXT I
1300 CALL KEY(0,K,S):: IF K<
>68 AND K<>83 THEN 1360
1310 IF N<11 THEN 1300
1320 IF K=68 THEN Z=Z+1 ELSE
 Z=Z-1
1330 IF Z=0 THEN Z=INT((N-1)
/10)+1
1340 IF Z>INT((N-1)/10)+1 TH
EN Z=1
1350 GOTO 1270
1360 IF K=2 THEN 1620
1370 IF K-47+(Z-1)*10>N THEN
 1300
1380 IF K<48 OR K>57 THEN 13
00
1390 IF I1=0 THEN I1=K-47+(Z
-1)*10 ELSE I2=K-47+(Z-1)*10
1400 IF SEG$(N$(I1),16,1)="*
" THEN I1=0 :: GOTO 1300
1410 IF SEG$(N$(I2),16,1)="*
" THEN I2=0 :: GOTO 1300
1420 DISPLAY AT(5,1):N$(I1)
1430 IF I2=0 OR I1=I2 THEN 1
300
1440 DISPLAY AT(6,1):"GEGEN"
:N$(I2)
1450 N$(I1)=N$(I1)&" *"
1460 N$(I2)=N$(I2)&" *"
1470 ACCEPT AT(5,17)VALIDATE
(DIGIT)SIZE(-2):Y$ :: IF Y$=
"" THEN 1470 ELSE X=VAL(Y$)
1480 ACCEPT AT(7,17)VALIDATE
(DIGIT)SIZE(-2):Y$ :: IF Y$=
"" THEN 1480 ELSE Y=VAL(Y$)
1490 IF X<10 AND Y<10 THEN 1
510
1500 DISPLAY AT(9,1):"ZWEIST
ELLIG ? N" :: ACCEPT AT(9,15
)VALIDATE("JN")SIZE(-1):Y$ :
: IF Y$="N" THEN 1470
1510 CALL HCHAR(9,3,32,15)
1520 T(I1)=T(I1)+X :: G(I1)=
G(I1)+Y
1530 T(I2)=T(I2)+Y :: G(I2)=
G(I2)+X
1540 IF X<=Y THEN 1560
1550 GE(I1)=GE(I1)+1 :: VERL
(I2)=VERL(I2)+1 :: GOTO 1590
1560 IF X<>Y THEN 1580
1570 UN(I1)=UN(I1)+1 :: UN(I
2)=UN(I2)+1 :: GOTO 1590
1580 GE(I2)=GE(I2)+1 :: VERL
(I1)=VERL(I1)+1
1590 I=I1 :: GOSUB 2430
1600 I=I2 :: GOSUB 2430
1610 CALL HCHAR(5,1,32,96)::
 I1,I2=0 :: GOTO 1270
1620 FOR I=1 TO N :: N$(I)=S
EG$(N$(I),1,14):: NEXT I
1630 GOTO 140
1640 REM * DATEN LADEN
1650 CALL CLEAR
1660 PRINT "LADEN VON DATEN
VON KASSET- TE":"==========
================":""
1670 DISPLAY AT(24,1):"SOLL
GELADEN WERDEN ? N" :: ACCEP
T AT(24,23)VALIDATE("JN")SIZ
E(-1):Y$
1680 IF Y$<>"J" THEN 140
1690 OPEN #1:"CS1",SEQUENTIA
L,INTERNAL,INPUT ,FIXED 192
1700 INPUT #1:N,LIGA$
1710 FOR I=1 TO N STEP 6
```

ABO SERVICE-KARTE

TI 5/86

Coupon

Ja, ich möchte von Ihrem Angebot Gebrauch machen.

Bitte senden Sie mir bis auf Widerruf ab sofort jeweils die nächsten zwölf Ausgaben an untenstehende Anschrift. Sollte ich nicht vier Wochen vor Ablauf schriftlich kündigen, läuft diese Abmachung automatisch weiter.

Name ____________________

Vorname ____________________

Straße/Hausnr. ____________________

Plz/Ort ____________________

Ich bezahle:

☐ per beiliegendem Verrechnungsscheck

☐ gegen Rechnung

☐ bargeldlos per Bankeinzug von meinem Konto

bei (Bank) und Ort ____________________

Kontonummer ____________________

Bankleitzahl ____________________

(steht auf jedem Kontoauszug)

Unterschrift ____________________

Von meinem Widerspruchsrecht habe ich Kenntnis genommen.

Unterschrift ____________________

Ich nehme zur Kenntnis, daß die Belieferung erst beginnt, wenn die Abo-Gebühr dem Verlag zugegangen ist.

TI REVUE

Abo-Service **5/86**
Postfach 1107
8044 UNTERSCHLEISSHEIM

PROGRAMMSERVICE

Hiermit bestelle ich in Kenntnis Ihrer Verkaufsbedingungen

die Listings dieses Heftes auf

☐ Diskette zum Preis von (25 DM)

☐ Kessette zum Preis von (10 DM)

Ich zahle:

Bar – per beigefügtem Geld ( )

per beigefügtem Scheck ( )

Gegen Bankebbuchung am Versandtag ( )

Zutreffendes bitte ankreuzen!

Meine Bank (mit Ortsname) ....

Meine Kontonummer ....

Meine Bankleitzahl .... (steht euf jedem Bankauszug)

Vorname .... Nechname ....

Str./Nr. .... PLZ / Ort ....

Hiermit bestätige ich mit meiner Unterschrift, Ihre Verkaufsbedingungen gelesen zu haben und zu akzeptieren.

Unterschrift ....

Bitte ausschneiden und einsenden an
TI-REVUE
KASSETTENSERVICE **5/86**
Postfach 1107
8044 Unterschleißheim

ANZEIGENSERVICE

Die große Börse für jeden Zweck in der TI REVUE. Kostenlos für Privat-Inserenten. Spottbillig für gewerbliche Anbieter. Einfach Coupon ausschneiden, fotokopieren o.ä., ausfüllen und ab die Post — Freimachen nicht vergessen! — Unsere Adresse steht auf dem Coupon, ebenso die Preise für gewerbliche Anbieter! Achtung! Wir weisen ausdrücklich darauf hin, daß wir offensichtlich gewerbliche Anzeigen nicht kostenlos veröffentlichen und uns jedweden Abdruck kostenloser Anzeigen vorbehalten müssen, insbesondere, wenn deren Inhalt nicht TI-typisch ist oder gegen geltendes Recht verstößt. Private Chiffreanzeigen werden nicht aufgenommen. Für Privatanbieter: maximal acht Zeilen à 28 Anschläge. Für gewerbliche Anbieter: 5 DM p. mm.

TI
Anzeigenabteilg.
Postfach 1107
8044 Lohhof

Name ______________________
Vornarne ______________________
Straße/Hausnr. ______________________
PLZ/Ort ______________________

RESERVIERUNGS-SERVICE

Hiermit bestelle ich in Kenntnis Ihrer Verkaufsbedingungen

( ) Exemplar TI SPECIAL 14,80. ( ) Exemplar **TI ASSEMBLER SPECIAL 19,80.**

Zutreffendes bitte ankreuzen!
Ich zahle:
per beigefügtem Scheck / Schein ( )
Gegen Bankabbuchung am Versandtag ( )
Meine Bank (mit Ortsname) .... (4)
Meine Kontonummer ....
Meine Bankleitzahl .... (steht auf jedem Bankauszug)
Vorname .... Nachname ....
Str./Nr. .... PLZ / Ort ....
Hiermit bestätige ich mit meiner Unterschrift, Ihre Verkaufsbedingungen gelesen zu haben und zu akzeptieren.

Unterschrift ....

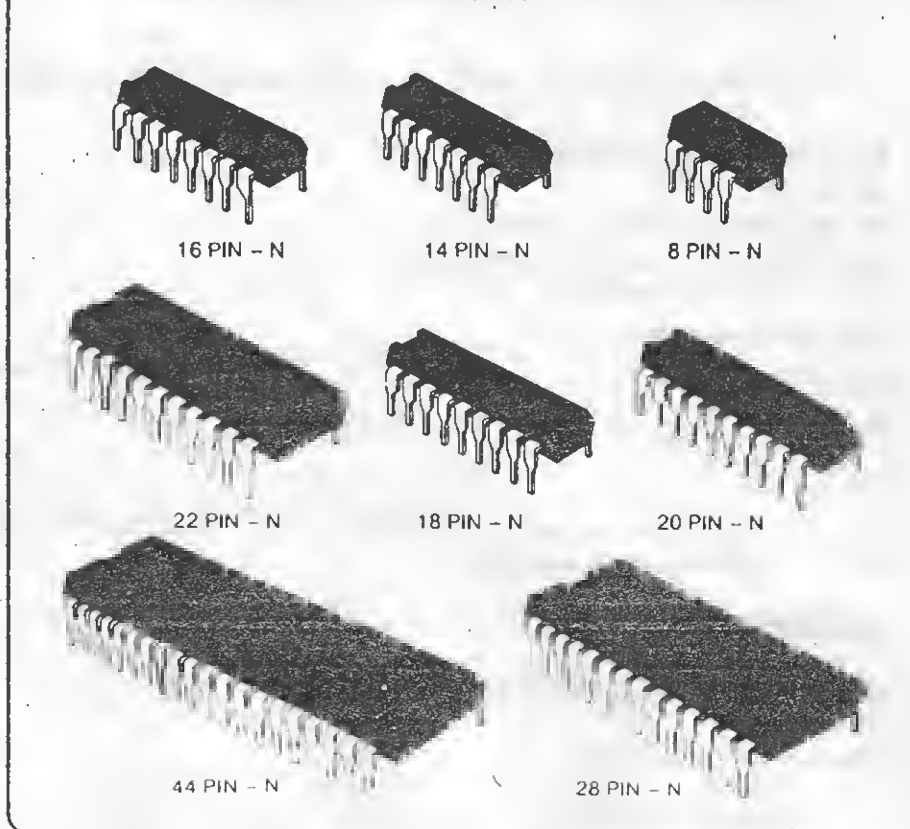
Bitte ausschneiden und einsenden an
TI-REVUE
Special-Service **5/86**
Postfach 1107
8044 Unterschleißheim





Gehäuse und thermische Angaben

PLASTIK



LISTINGS

```
1720 INPUT #1:TORE(I),N$(I),
BILANZ(I),TORE(I+1),N$(I+1),
BILANZ(I+1),TORE(I+2),N$(I+2
).BILANZ(I+2).
1730 INPUT #1:TORE(I+3),N$(I
+3),BILANZ(I+3),TORE(I+4),N$
(I+4),BILANZ(I+4),TORE(I+5),
N$(I+5),BILANZ(I+5)
1740 NEXT I
1750 CLOSE #1
1760 GOTO 140
1770 REM * DATEN SPEICHERN
1780 CALL CLEAR
1790 PRINT "SPEICHERN VON DA
TEN AUF":"KASSETTE":"=======
=====================":
1800 DISPLAY AT(24,1):"SOLL
GESPEICHERT WERDEN ? N" :: A
CCEPT AT(24,27)VALIDATE("JN"
)SIZE(-1):Y$
1810 IF Y$<>"J" THEN 140
1820 PLUS,MINUS,ETOR,GTOR=0
:: FOR I=1 TO N :: PLUS=PLUS
+P(I):: MINUS=MINUS+M(I):: E
TOR=ETOR+T(I):: GTOR=GTOR+G(
I):: NEXT I
1830 IF PLUS=MINUS AND ETOR=
GTOR THEN 1890
1840 PRINT "":"":"" :: DISPL
AY AT(22,1):"FEHLER BEI DEN
WERTEN":"SOLL TROTZDEM ABGES
PEICHERT WERDEN? N"
1850 ACCEPT AT(24,9)VALIDATE
("JN")SIZE(-1):Y$ :: IF Y$<>
"JJ" THEN 140
1860 FOR I=1 TO N
1870 GOSUB 2430
1880 NEXT I
1890 OPEN #1:"CS1",SEQUENTIA
L,INTERNAL,OUTPUT,FIXED 192
1900 PRINT #1:N,LIGA$
1910 FOR I=1 TO N STEP 6
1920 PRINT #1:TORE(I),N$(I),
BILANZ(I),TORE(I+1),N$(I+1),
BILANZ(I+1),TORE(I+2),N$(I+2
),BILANZ(I+2),
1930 PRINT #1:TORE(I+3),N$(I
+3),BILANZ(I+3),TORE(I+4),N$
(I+4),BILANZ(I+4),TORE(I+5),
N$(I+5),BILANZ(I+5)
1940 NEXT I
1950 CLOSE #1
1960 GOTO 140
1970 REM * TOR-STATISTIK
1980 IF LIGA$="" THEN DISPLA
Y AT(24,7):"KEINE DATEN !!!!
" :: GOTO 190
1990 DISPLAY AT(4,1)ERASE AL
L:"STATISTIK" :: GOSUB 2260
:: ETOR,SPI=0
2000 FOR L=1 TO N
2010 GOSUB 2450
2020 SP(L)=P(L)/2+M(L)/2
2030 NEXT L
2040 FOR I=1 TO N
2050 ETOR=ETOR+T(I)
2060 SPI=SPI+SP(I)
2070 NEXT I
2080 DUM=ETOR/N
2090 SPI=SPI/2 :: DUSP=(INT(
10*ETOR/SPI))/10
2100 CALL CLEAR :: DISPLAY A
T(1,1):USING "TORE:.........
..........####DURCHSCHNITT M
ANNSCHAFT: ###DURCHSCNNITT S
PIEL:.....##.#":ETOR,DUM,DUS
P
2110 CALL HCHAR(4,1,61,32)::
 DISPLAY AT(5,1):"MANNSCHAFT
......SP. ...TORE" :: J=0 ::
 FOR I=1 TO N :: J=J+1
2120 IF J<19 THEN 2140
2130 J=18 :: PRINT
2140 DISPLAY AT(5+J,1):USING
 "#############...##..###:#
##":N$(I),SP(I),T(I),G(I)
2150 NEXT I
2160 CALL KEY(0,K,S):: IF S<
>0 THEN 140 ELSE 2160
2170 REM * ENDE
2180 DISPLAY AT(11,11)ERASE
ALL:"E N D E"
2190 FOR I=1 TO 75
2200 CALL KEY(0,K,S):: IF S<
>0 THEN 140
2210 NEXT I :: CALL CLEAR ::
 END
2220 REM * UNTERPROGRAMME
2230 REM * BILDSCHIRM LOE-
           SCHEN UND PUNKTE
           DEFINIERT
2240 CALL HCHAR(1,1,95,768):
: CALL CHAR(32,"000000000000
6666"):: RETURN
2250 REM * UNTERSTREICHEN
2260 CALL HCHAR(5,1,61,32)::
 RETURN
2270 REM * VERTAUSCHEN
2280 F=1 :: H=P(I):: P(I)=P(
I+1):: P(I+1)=H
2290 H=M(I):: M(I)=M(I+1)::
M(I+1)=H
2300 H=BILANZ(I):: BILANZ(I)
=BILANZ(I+1):: BILANZ(I+1)=H
2310 H=T(I):: T(I)=T(I+1)::
T(I+1)=H
2320 H=G(I):: G(I)=G(I+1)::
G(I+1)=H
2330 H=TG(I):: TG(I)=TG(I+1)
:: TG(I+1)=H
2340 H=TORE(I):: TORE(I)=TOR
E(I+1):: TORE(I+1)=H
```

```
2350 H1$=N$(I):: N$(I)=N$(I+
1):: N$(I+1)=H1$
2360 RETURN
2370 REM * DATEN LOESCHEN
2380 FOR I=1 TO N
2390 N$(I)=""
2400 TORE(I),T(I),G(I),TG(I)
,BILANZ(I),GE(I),UN(I),VERL(
I),SP(I),P(I),M(I)=0
2410 NEXT I :: RETURN
2420 REM * ZAHLENUMWANDLUN-
        GEN
2430 TORE(I)=T(I)*1000+G(I)
2440 BILANZ(I)=GE(I)*10000+U
N(I)*100+VERL(I):: RETURN
2450 T(I)=INT(TORE(I)/1000):
: G(I)=TORE(I)-T(I)*1000 ::
TG(I)=T(I)-G(I)
2460 GE(I)=INT(BILANZ(I)/100
00):: UN(I)=INT(BILANZ(I)/10
0)-GE(I)*100 :: VERL(I)=BILA
NZ(I)-GE(I)*10000-UN(I)*100
2470 P(I)=GE(I)*2+UN(I):: M(
I)=UN(I)+VERL(I)*2 :: RETURN
```

SIMPLEXTABLEAU

Fortsetzung von Seite 26

Tableaus an, so bemerkt man, daß hier die Dekkungsbeiträge der einzelnen Produkte negativ eingetragen sind, und zwar wegen der Umformung der Gleichung DB=$30x_1 + 90x_2$, da im Tableau das '='-Zeichen zwischen letzte und vorletzte Spalte gedacht werden muß. Daher wird der Computer in der ersten Zeile nie Werte größer als Null annehmen. Die einzelnen Maschinenengpässe werden als Ungleichungen mathematisch dargestellt und sodann als Gleichung ins Tableau geschrieben.
So ist im Falle der Maschine A die maximale Kapazität bei 450 Stück, Produkt 1 benötigt pro Stück 6 Stunden, Produkt 2 pro Stück 9 Stunden. Die Ungleichung lautet daher:

$$6x_1 + 9x_2 \leq 450$$

Ähnlich funktioniert dies mit Maschine B.
Die Absatzbeschränkung von 2 ist mit 40 angegeben, es wird daher in der 2er-Spalte eine eins und in der 1er-Spalte eine Null eingetragen, da Produkt 1 nicht davon betroffen ist.
Nach dieser Vorgangsweise wird das Ausgangstableau festgelegt, welches nun in den Computer eingegeben werden muß. Nach Abschluß der Eingabetätigkeit wurde ein reales Problem in einem mathematischen Modell abgebildet, was das Wesen der Operations Research ist.
Um Ihnen das Eintippen der Beispielangaben zu ersparen, erfolgt die Zuweisung zu den entsprechenden Variablen durch Wahl der Option 2.
Aus dem Anhang ersehen Sie die Durchführung der iterativen Schritte, die der Computer durchführt, bis er den optimalen Dekkungsbeitrag errechnet hat. Hier sind es 2 Schritte.

3. Erläuterungen zum Programm:

Notwendige Konfiguration: TI 99/4A Konsole, CS1, X.-Bas.-Modul

Nach dem Programmstart in Extended Basic mit 'RUN' muß zwischen '1'

Bitte lesen Sie weiter auf Seite 39

LISTINGS

```
10 ! **********************
11 ! *  SIMPLEX-TABLEAU    *
12 ! *     (Vers. 3.2.)    *
13 ! *                     *
14 ! *    Copyright by     *
15 ! *                     *
16 ! *  N. Kaltenbrunner   *
17 ! *                     *
19 ! * Benoetigte Geraete  *
20 ! *  TI99/4A  Konsole   *
21 ! *     Ext. Basic      *
22 ! *                     *
26 ! *  Speicherbelegung   *
27 ! *     5636 Bytes      *
28 ! *                     *
29 ! **********************
100 CALL CLEAR :: CALL SCREE
N(4):: OPTION BASE 1 :: CALL
 CHAR(123,"0010101010101000"
):: CALL COLOR(3,2,12,4,2,12
)
110 DIM Z(20,20),W(20,20),Q(
20),PR$(20)
120 B=0 :: SCHL=0
130 EMAX=20 :: PMAXI=20
140 PRINT "..***************
*******......*SIMPLEX-TABLEA
U..3.2*......***************
*******"
150 PRINT :"....(c)..1985..b
y..Nk": : :
160 INPUT "..1=Simplex - neu
e Eingabe....2=Demonstration
sbeispiel...................
................":ENT
170 IF ENT=1 THEN 190
180 GOSUB 1910 :: GOTO 440
190 INPUT "Wieviele Produkte
? ":P :: IF P<1 OR P>PMAXI T
HEN 190
200 FOR J=2 TO P+1
210 PRINT "Produktbezeichnun
g";J-1;"?"
220 INPUT PR$(J):: IF LEN(PR
$(J))>3 THEN 210
230 NEXT J
240 INPUT "Wieviele Engpaess
e? ";E :: IF E<1 OR E>EMAX T
HEN 240
250 FOR I=2 TO E+1
260 PRINT "Engpassbezeichnun
g";I-1;"?"
270 INPUT B$(I):: IF LEN(B$(
I))>3 THEN 260
280 NEXT I
290 CALL CLEAR :: PRINT "**E
ingabe Zielfunktion**" :: PR
INT
300 I=1
310 FOR J=2 TO P+1
320 PRINT "DB Produkt";J-1;P
```

```
R$(J)
330 INPUT Z(I,J):: IF Z(I,J)
>0 THEN 320
340 NEXT J
350 CALL CLEAR :: PRINT "**E
ingabe Engpaesse**": :
360 FOR I=2 TO E+1 :: SUM=0
370 FOR J=2 TO P+1
380 PRINT "Eingabe Engpasswe
rt fuer....Engpass: ";B$(I);
" Produkt: ";PR$(J)
390 INPUT Z(I,J):: SUM=SUM+Z
(I,J)
400 NEXT J
410 PRINT "Gesamtkapazitaet
des Eng-...passes";B$(I);"?"
420 INPUT Z(I,P+2):: IF Z(I,
P+2)<SUM THEN PRINT "Das ist
 unmoeglich, Zeile   neu ein
geben!" :: SUM=0 :: GOTO 370
430 NEXT I
440 CALL CLEAR :: INPUT "Ber
echnung aller Schritte...auf
 einmal (1) oder Ausgabe jed
es einzelnes Tableaux´(2)":G
ESAMT
450 Z(1,P+2)=0 :: INPUT "...
...........................Kon
trollausgabe?(J/N) ":A$
460 IF A$<>"J" THEN 470 ELSE
 GOSUB 1430
470 PR$(1)="0" :: PR$(P+2)="
OPT" :: B$(1)="DB"
480 CALL SCREEN(14):: CALL H
CHAR(12,1,32,32):: DISPLAY A
T(13,1):".....Ich rechne....
........." :: CALL HCHAR(14,
1,32,32)! Suche DB min
490 Z(1,1)=0
500 FOR I=2 TO E+1
510 Z(I,1)=1
520 NEXT I
530 MINQ=0
540 SCHL=SCHL+1
550 I=1
560 J=2
570 S=Z(I,J)
580 MINJ=2
590 FOR J=3 TO P+1
600 IF Z(I,J)<S THEN 630
610 NEXT J
620 GOTO 670
630 S=Z(I,J)
640 MINJ=J
650 GOTO 590
660 ! MINJ=niedr.Spalte
670 J=MINJ
680 ! Suche min Quotient
690 FOR I=2 TO E+1
700 IF Z(I,J)=0 THEN 730
710 Q(I)=Z(I,P+2)/Z(I,J)
720 GOTO 740
730 Q(I)=99999999 ! entspric
ht unendlich
740 NEXT I
750 I=2
760 MINQ=Q(I)
770 MINZ=I
780 FOR I=3 TO E+1
790 IF Q(I)<MINQ THEN 820
800 NEXT I
810 GOTO 860
820 MINQ=Q(I)
830 U=I
840 MINZ=U
850 GOTO 800
860 ! Pivotelement ist in Ze
ile MINZ Spalte MINJ und hat
den WERT Z(MINZ,MINJ)
870 PIVZ=MINZ :: PIVS=MINJ
880 ! Umrechnen
890 T$=PR$(PIVS)
900 PR$(PIVS)=B$(PIVZ)
910 B$(PIVZ)=T$
920 ! Pivotzeile bestimmen
930 GOTO 940
940 IF Z(PIVZ,PIVS)=1 THEN 1
000
950 R=Z(PIVZ,PIVS)
960 FOR J=1 TO P+2
970 Z(PIVZ,J)=Z(PIVZ,J)/R
980 NEXT J
990 B=0
1000 ! Zeile aendern
1010 B=B+1
1020 IF B=PIVZ THEN 1040
1030 GOTO 1050
1040 B=B+1
1050 FOR J=1 TO P+2
1060 W(PIVZ,J)=Z(PIVZ,J)
1070 NEXT J
1080 D=Z(B,PIVS)
1090 GOSUB 1200
1100 IF B>E THEN 1120
1110 GOTO 1000
1120 D=Z(PIVZ,PIVS)
1130 Z(PIVZ,PIVS)=Z(PIVZ,1)
1140 Z(PIVZ,1)=D
1150 IF GESAMT=1 THEN 1800
1160 GOSUB 1430
1170 IF WEITER=1 THEN 480
1180 CALL HCHAR(16,1,32,32):
: CALL HCHAR(18,1,32,32):: D
ISPLAY AT(17,1):"Neue Eingab
e? (J/N)" :: ACCEPT AT(17,22
):A$ :: IF A$="N" THEN END E
LSE GOTO 100
1190 GOTO 100
1200 ! Zeilenaenderung
1210 FOR J=1 TO P+2
1220 Z(B,J)=Z(B,J)*W(PIVZ,PI
VS)
```

LISTINGS

```
1230 NEXT J
1240 IF Z(B,PIVS)=1 THEN 130
0
1250 ! Pivotzeile aendern
1260 FOR J=1 TO P+2
1270 Z(PIVZ,J)=Z(PIVZ,J)*D
1280 NEXT J
1290 ! Subtraktion
1300 FOR J=1 TO P+2
1310 Z(B,J)=Z(B,J)-Z(PIVZ,J)
1320 NEXT J
1330 D=Z(B,1)
1340 Z(B,1)=Z(B,PIVS)
1350 Z(B,PIVS)=D
1360 FOR J=1 TO P+2
1370 Z(PIVZ,J)=W(PIVZ,J)
1380 NEXT J
1390 IF B=1 THEN 1420
1400 IF Z(B,PIVS)=0 THEN 142
0
1410 Z(B,PIVS)=Z(B,PIVS)+(-1
)
1420 RETURN
1430 DIF=4 :: M=1 :: N=2 ::
WEITER=0 ! Ausgabe
1440 CALL CLEAR :: CALL SCRE
EN(4):: Z2=3 :: S2=7
1450 FOR J=2+N-2 TO P+2
1460 DISPLAY AT(1,S2):PR$(J)
1470 S2=S2+DIF :: IF S2>20 T
HEN 1490
1480 NEXT J
1490 S2=7 :: CALL HCHAR(2,4,
45,28)
1500 FOR I=M TO E+1
1510 DISPLAY AT(Z2,2)SIZE(DI
F):B$(I)
1520 Z2=Z2+1 :: IF Z2>19 THE
N 1540
1530 NEXT I
1540 Z2=3 :: CALL VCHAR(1,7,
123,19)
1550 FOR I=M TO E+1
1560 S2=7 :: FOR J=2+N-2 TO
P+2
1570 DISPLAY AT(Z2,S2):Z(I,J
)
1580 S2=S2+DIF :: IF S2>20 T
HEN 1600
1590 NEXT J
1600 Z2=Z2+1 :: IF Z2>19 THE
N 1620
1610 NEXT I
1620 FOR J=2 TO P+1 :: IF Z(
1,J)<0 THEN WEITER=1
1630 NEXT J
1640 DISPLAY AT(21,1):"*****
*************************Ente
r:weiter A:Spa.abstand**E,S,
D,X W:Matrix I:Interpr*"
1650 IF WEITER=0 THEN DISPLA
Y AT(21,1):"**=> Optimales E
rgebnis <=**" :: CALL SOUND(
220,220,3):: CALL SOUND(220,
440,3):: CALL SOUND(220,880,
3)
1660 CALL KEY(0,KK,SS)
1670 IF KK=65 THEN 1770
1680 IF KK=73 AND WEITER=0 T
HEN 1840
1690 IF KK=88 THEN M=M-1 ::
IF M<1 THEN M=1
1700 IF KK=69 THEN M=M+1 ::
IF M>E+1 THEN M=E+1
1710 IF KK=83 THEN N=N-1 ::
IF N<2 THEN N=2
1720 IF KK=68 THEN N=N+1 ::
IF N>P+2 THEN N=P+2
1730 DISPLAY AT(24,1):"*Ze:"
;M;"Sp:";N-1;"Matrix";SCHL :
: DISPLAY AT(24,28):"*"
1740 IF KK=87 THEN 1440
1750 IF KK=13 THEN RETURN
1760 GOTO 1660
1770 ! Spaltenabstand
1780 DISPLAY AT(21,1):"*=>Ne
uer Spaltenabstand?" :: ACCE
PT AT(21,26):DIF :: IF DIF>1
5 OR DIF<4 THEN 1780
1790 GOTO 1440
1800 FOR J=2 TO P+1
1810 IF Z(1,J)<0 THEN 480
1820 NEXT J
1830 GOTO 1160
1840 CALL CLEAR :: CALL SCRE
EN(16):: PRINT "...=========
=====..............Interpret
ation..............=========
=====....................."
1850 PRINT "Optimales Ergebn
is (Z.B. im Beispiel max. De
ckungsbei-..tragssumme):";Z(
1,P+2)
1860 PRINT "In Zeile 1 stehe
n unter Pro-duktbezeichnunge
n die Grenz-nutzensaetze.Bei
 Beschraen- kungen bedeutet
dies, dass..deren Kapazitaet
 ausgelastetist."
1870 PRINT "Ab der Zeile 2 g
ilt:"
1880 PRINT :: PRINT "In den
Zeilen, wo links die Produkt
bezeichnungen stehen,sind ga
nz rechts die optimalzu prod
uzierenden Stueckzah-Ien zu
finden."
1890 PRINT "Bei Engpaessen s
tehen dort..die noch freien
Kapazitaeten"
1900 CALL KEY(0,K,S):: IF S<
>0 THEN 1430 ELSE 1900
```

```
1910 RESTORE ! Beispiel
1920 FOR I=1 TO 4 :: FOR J=2
 TO 4 :: READ Z(I,J):: NEXT
J :: NEXT I :: READ PR$(1),P
R$(2),PR$(3),PR$(4),B$(1),B$
(2),B$(3),B$(4):: READ E,P
1930 !EAD E,P
1940 DATA -30,-90,0
1950 DATA 6,9,450
1960 DATA 8,6,480
1970 DATA 0,1,40
1980 DATA NU,X1,X2,OPT,DB,A,
B,P
1990 DATA 3,2
2000 RETURN
```

LISTINGS

SIMPLEXTABLEAU

Fortsetzung von Seite 36

(Eintippen von Angaben) und '2' (Daten für Demonstrationsbeispiel einlesen) gewählt werden. Darauf wird abgefragt, ob alles auf einmal (1) oder ob die Berechnung von Zwischentableau zu Zwischentableau erfolgen soll. Mit einer Kontrollausgabe kann überprüft werden, ob richtig eingetippt wurde.

– Bildschirmausgabe:
Die Ausgabe ist so konzipiert, daß der Bildschirm praktisch eine Lupe darstellt, die mit Hilfe der Pfeiltasten über das Arbeitsblatt der Matrix geführt wird. Die Zeilen- u. Spaltenangaben beziehen sich auf die linke oberste Zahl der Matrix. Sollte der Spaltenabstand nicht ausreichen, dann kann dieser mit Taste 'A' verändert werden. Mit 'W' wird die Matrix ohne Weiterrechnung neu am Schirm ausgegeben. Mit 'ENTER' erfolgt die Berechnung des nächsten Schrittes bzw. des Endergebnisses.
Ist das optimale Ergebnis erreicht, dann ertönt ein akustisches Signal, die Wahl der Option 'I' wird möglich. Diese bringt am Schirm eine kleine Interpretation des Ergebnisses. Mit diesem Programm können max. 20 Gleichungen verarbeitet werden. Dabei ist zu beachten, daß es sich um ein lineares Gleichungssystem handelt. Ist ein Index eines Matrizenwertes Null, dann erfolgt ein Absturz.
Durch Herauslöschen der Musterbeispielangaben bzw. der Kommentierung der wichtigsten Variablen (Zeile 117 – 169) kann Speicherplatz gewonnen werden.
Das Programm wurde zwar in Extended-Basic geschrieben, trotzdem wurden die Fähigkeiten dieses Dialekts hauptsächlich in der Ausgaberoutine verwertet, um Usern anderer Computertypen einen Einblick in den Algorithmus zu ermöglichen.

Beispiel:

Ein Betrieb erzeugt die Produkte P1 und P2 auf den Maschinen A und B. Produkt P1 (P2) benötigt zu seiner Fertigung auf Maschine A 6 (9) Stunden und auf Maschine B 8 (6) Stunden. Von der Maschine A (B) stehen in der Periode maximal 450 (480) Stunden zur Verfügung.
Der Absatz von P2 ist auf 40 Stück beschränkt.
Mit P1 (P2) kann ein Stückdeckungsbeitrag von 30,– (90,–) erzielt werden.

Aufgabenstellung:

Bestimmen Sie das deckungsbeitragsmaximale Produktionsprogramm!

Zeilennummernbedeutung im Programm Simplextableau:

100–410
Initialisierung
420–690
Angabe eintippen
710–1420
Schleife zur Berechnung eines Tableaus
1230–1340
Schleife fürs Zeilenändern im jeweiligen Tableau
1430–1650
Unterprogramm Zeilenänderung
1660–2020
UP Bildschirmausgabe
2030–2060
UP Überprüfung, ob optimales Ergebnis erreicht
2070–2130
UP Ergebnisinterpretation
2140–2230
UP Angaben des Musterbeispiels

LERN-PARTNER

Lernpartner ist ein leistungsfähiges Lernprogramm für universelle Anwendungen. Es bietet Hilfe und Unterstützung beim Auswendiglernen kleinerer Texteinheiten, beim Lernen von Vokabeln oder mathematischen–naturwissenschaftlichen Formeln und Lehrsätzen, bei Rechtschreibübungen.
Die Programmgestaltung entspricht den natürlichen Eigenschaften des menschlichen Gedächtnisses und seines Lernvermögens:

- Der Computer verlangt für die Lernkontrollen die schriftliche Eingabe, was den Lernprozeß fördert, und die effektive Lernzeit erheblich reduziert.
- Lektion 1 (Text lernen) stellt jeweils nur einen kurzen Textabschnitt zum Lernen vor. Danach muß blind die Eingabe erfolgen, und das Programm führt die Kontrollen durch. Das Gelernte – zunächst nur im Kurzzeitgedächtnis gespeichert! – wird dabei vertieft, und die Übertragung ins Langzeitgedächtnis erleichtert.
- Lektion 2 (Text lernen) verlangt sofort die blinde Eingabe. So wird kontrolliert, ob das Gelernte wirklich korrekt im Langzeitgedächtnis gespeichert wurde.
- Die Lektionen lassen sich beliebig oft wiederholen.

Das Programm ist menügesteuert und arbeitet mit Bildschirmmasken. Es erlaubt Groß- und Kleinschrifteingaben, sowie Umlaute auf den üblichen Tasten (FCTN+ R,Z,T,F, A,G,W). Im Texteingabemodus finden die Tasten INSERT, DELETE, ERASE, ARROW RIGHT und ARROW LEFT ihre vom BASIC-Befehl ACCEPT AT her bekannten Verwendungen. Die Anwendungen von ENTER, PROC'D, AID, BACK, ERASE und QUIT werden zudem im Programm erläutert.
Bei den Texteingaben ist es empfehlenswert, sich eine bestimmte Technik anzugewöhnen:

- a) Schriftart: Nur Großschrift oder Groß-/Kleinschrift
- b) Nutzung des Zeilenendes: Überschreiben des Zeilenendes ohne Trennstrich und grammatisch richtige Form, mit Worttrennung in richtiger Form am Zeilenende, ohne Worttrennung (Zeilenabbruch, wenn das folgende Wort die Zeile überschreiben würde).

Diese Gewöhnung erspart Probleme in der Lernkontrolle, denn das Programm unterscheidet natürlich genau zwischen "AUTO" und "Auto" in der Rechtschreibprüfung.

Bitte lesen Sie weiter auf Seite 42

LISTINGS

```
10 REM********************
11 REM* GARTEN DER PILZE *
12 REM*                 *
13 REM*  Copyright  by   *
14 REM*                 *
15 REM*Oliver Wallenfang *
16 REM*                 *
17 REM*Benoetigte Geraete*
18 REM* TI99/4A  Konsole *
19 REM*     Joystick     *
20 REM*                 *
23 REM* Speicherbelegung *
24 REM*    6402 Bytes    *
25 REM********************
26 REM
100 CALL CLEAR
110 GOSUB 580
120 PRINT TAB(5);"**********
*********":TAB(5);"*.......
...........*":TAB(5);"* GART
EN DER PILZE *"
130 PRINT TAB(5);"*.........
.........*":TAB(5);"********
***********": : :TAB(5);"SP
IELREGELN ?": :TAB(5);"(j/n)
": : : :
140 CALL KEY(0,K,S)
150 IF S=0 THEN 140
160 IF K=110 THEN 810
170 IF K=106 THEN 230
180 CALL SOUND(-200,-1,0)
190 IF K>96 THEN 140
200 PRINT "<ALPHA LOCK>-TAST
E LOESEN!"
210 GOTO 140
220 REM ANLEITUNG
230 PRINT TAB(5);"** GARTEN
DER PILZE **": : :
240 PRINT "SIE SIND VON EINE
M GRAUSAMEN": :"HERRSCHER GE
FANGENGENOMMEN,": :"DER SIE
DAZU ZWINGT,PILZE": :
250 PRINT "AUS EINEM,VON EIN
EM GESPENST": :"BEWACHTEN,GA
RTEN ZU SAMMELN.": : :"ERST
WENN SIE GENUEGEND": :
260 PRINT "PILZE HABEN,LAESS
T ER DIE": :"MAUER ENTFERNEN
 UND SIE": :"FLUECHTEN.": :T
AB(5);">>"
270 CALL KEY(0,K,S)
280 IF S=0 THEN 270
290 CALL CLEAR
300 PRINT "BEI JEDEM PILZ DE
N SIE": :"SAMMELN VERLIEREN
SIE": :"GIFTPILZSPOREN,DIE,W
ENN SIE": :
310 PRINT "MIT EINEM MENSCHL
ICHEN WESEN": :"IN BERUEHRUN
G KOMMEN,SOFORT": :"ZU RIESI
GEN PILZEN WERDEN,": :
320 PRINT "DIE IHNEN DEN WEG
 VERSPERREN": :"KOENNEN.": :
"DEN WEG VERSPERREN KOENNEN"
: :"AUCH DIE BUESCHE,DIE": :
330 PRINT "UEBERALL IM GARTE
N VERSTREUT": :"STEHEN.":TAB
(10);">>";
340 CALL KEY(0,K,S)
350 IF S=0 THEN 340
360 CALL CLEAR
370 PRINT "DIE BUESCHE UND P
ILZE VER-": :"SPERREN DEM GE
SPENST DEN": :"WEG.ALLERDING
S KANN ES": :
380 PRINT "MANCHMAL EIN HIND
ERNIS": :"UEBERSPRINGEN": :
:"WENN SIE DIE PILZE": :"ABG
ELIEFERT HABEN,WERDEN": :
390 PRINT "SIE NOCH EINMAL L
OS-": :"GESCHICKT.": :"DIESM
AL ERHALTEN SIE ABER": :"ALS
 BELOHNUNG MEHR PUNKTE.": :
400 PRINT TAB(10);">>";
410 CALL KEY(0,K,S)
420 IF S=0 THEN 410
430 CALL CLEAR
440 PRINT "ZEICHEN :": : :"S
IE SELBST =";CHR$(127): :"GE
SPENST...=";CHR$(144): :"PIL
Z.......=";CHR$(128):
450 PRINT :"GIFTPILZ...=";CH
R$(152): :"BUESCHE....=";CHR
$(137);CHR$(138);CHR$(139)
460 PRINT TAB(10);">>"
470 CALL KEY(0,K,S)
480 IF S=0 THEN 470
490 CALL CLEAR
500 PRINT "GESTEUERT WIRD MI
T JOYST.1.": :"SIE MUESSEN A
N DEN LINKEN": :"BILDSCHIRMR
AND GELANGEN.": :
510 PRINT "DIE PUNKTE ENTSTE
HEN AUS DEN": :"PILZEN UND D
ER RUNDE.": :
520 PRINT "DER BONUS ERHOEHT
 SICH BEI": :"JEDER RUNDE UM
 5 PUNKTE.": :"JE NACH SCHWI
ERIGKEITSSTUFE": :
530 PRINT "MUESSEN SIE ZWISC
HEN": :"10 UND 50 PILZE SAMM
ELN.": :"DAS SPIEL BEGINNT M
IT DEM": :"FEUERKNOPF";
540 CALL KEY(0,K,S)
550 IF S=0 THEN 540
560 CALL SOUND(10,440,0)
570 GOTO 810
580 RANDOMIZE
590 REM ZEICHEN
600 CALL CHAR(128,"0000387C1
01")
610 CALL CHAR(127,"383810385
```

```
4102844")
620 CALL CHAR(144,"1C3E7F495
D7F7F55")
630 CALL CHAR(152,"003C5AF71
8181818")
640 CALL CHAR(137,"000C0E1EF
E4C6E30")
650 CALL CHAR(138,"206C3C36E
67C1008")
660 CALL CHAR(139,"001A7E22F
E143C0C")
670 CALL CHAR(119,"0076006E0
076006E")
680 FOR I=1 TO 10
690 CALL COLOR(I,16,1)
700 NEXT I
710 CALL COLOR(11,15,1)
720 CALL COLOR(12,16,1)
730 CALL COLOR(13,10,1)
740 CALL COLOR(14,3,1)
750 CALL COLOR(15,16,1)
760 CALL COLOR(16,7,1)
770 CALL CLEAR
780 CALL SCREEN(2)
790 RETURN
800 REM SCHW.-STUFEN
810 CALL CLEAR
820 PRINT "1____ANFAENGER":
:"2____FORTGESCHRITTEN": :"3
____PROFI": :"4____MEISTER":
 :"5____VERRUECKT": :TAB(10)
;
830 CALL KEY(0,K,S)
840 IF S=0 THEN 830
850 SCH=K-48
860 IF SCH<1 THEN 830
870 IF SCH>5 THEN 830
880 CALL CLEAR
890 DIM A(24,32)
900 REM MAUER
910 FOR I=1 TO 24
920 CALL HCHAR(I,4,119)
930 A(I,4)=1
940 CALL SOUND(1,-7,1)
950 NEXT I
960 REM BUESCHE
970 FOR I=1 TO 3
980 FOR B=5 TO 32
990 H=INT(24*RND)+1
1000 Z=INT(3*RND)+137
1010 IF A(H,B)<>0 THEN 990
1020 CALL HCHAR(H,B,Z)
1030 CALL SOUND(3,-3,1)
1040 A(H,B)=1
1050 NEXT B
1060 NEXT I
1070 REM PILZE
1080 FOR I=1 TO 2
1090 FOR B=5 TO 32
1100 H=INT(24*RND)+1
1110 IF A(H,B)<>0 THEN 1100
```

LISTINGS

```
1120 CALL HCHAR(H,B,128)
1130 CALL SOUND(1,-1,1)
1140 A(H,B)=2
1150 NEXT B
1160 NEXT I
1170 IF ZX=1 THEN 1210
1180 ZX=1
1190 REM VARIABLEN
1200 RU=1
1210 H=INT(24*RND)+1
1220 IF A(H,30)<>0 THEN 1210
1230 H1=H
1240 B=30
1250 B1=B
1260 CALL HCHAR(H,B,127)
1270 HG=INT(24*RND)+1
1280 IF A(HG,5)<>0 THEN 1270
1290 HG1=HG
1300 BG=5
1310 BG1=BG
1320 Z=0
1330 CALL HCHAR(HG,BG,144)
1340 CALL KEY(1,K,S)
1350 IF S=0 THEN 1340
1360 IF K<>18 THEN 1340
1370 CALL SOUND(50,440,0)
1380 REM BEWEGUNG SPIELFIGUR
1390 CALL JOYST(1,X,Y)
1400 IF (X=0)*(Y=0)THEN 1660
1410 H=H-(Y=-4)+(Y=4)
1420 B=B-(X=4)+(X=-4)
1430 H=H+(H>24)-(H<1)
1440 B=B+(B>32)-(B<1)
1450 IF B=1 THEN 2130
1460 IF A(H,B)<>3 THEN 1500
1470 A(H,B)=1
1480 CALL HCHAR(H,B,152)
1490 CALL SOUND(100,262,1,33
0,1,392,1)
1500 IF A(H,B)<>2 THEN 1560
1510 A(H,B)=3
1520 Z=Z+1
1530 CALL SOUND(50,262,1)
1540 CALL SOUND(100,349,0)
1550 IF Z=SCH*10 THEN 1820
1560 IF A(H,B)<>1 THEN 1600
1570 H=H1
1580 B=B1
1590 CALL SOUND(100,110,3)
1600 IF (H=H1)*(B=B1)THEN 16
50
1610 CALL HCHAR(H,B,127)
1620 CALL HCHAR(H1,B1,32)
1630 H1=H
1640 B1=B
1650 REM BEWEGUNG GEGNER
1660 HG=HG+(HG>H)-(HG<H)+(RN
D>RND*1.2)-(RND>RND*1.2)
1670 BG=BG+(BG>B)-(BG<B)+(RN
D>RND*1.2)-(RND>RND*1.2)
1680 HG=HG+(HG>24)-(HG<1)
```

```
1690 BG=BG+(BG>32)-(BG<1)
1700 IF BG<>B THEN 1720
1710 IF HG=H THEN 1890
1720 IF A(HG,BG)=0 THEN 1750
1730 HG=HG1
1740 BG=BG1
1750 IF (HG=HG1)*(BG=BG1)THE
N 1800
1760 CALL HCHAR(HG,BG,144)
1770 CALL HCHAR(HG1,BG1,32)
1780 HG1=HG
1790 BG1=BG
1800 GOTO 1390
1810 REM EINREISSEN DER MAUE
R
1820 FOR I=1 TO 24
1830 CALL HCHAR(I,4,32)
1840 A(I,4)=0
1850 CALL SOUND(10,-6,1)
1860 NEXT I
1870 GOTO 1560
1880 REM GEFANGEN !
1890 CALL HCHAR(HG,BG,144)
1900 CALL HCHAR(HG1,BG1,32)
1910 FOR I=0 TO 30
1920 CALL SOUND(-100,INT(500
*RND)+110,I,INT(500*RND)+112
,I)
1930 NEXT I
1940 CALL CLEAR
1950 PRINT "DAS GESPENST HAT
 SIE": :"ERWISCHT !": : :
1960 PRINT "PILZE.....:";Z:
:"RUNDE.....:";RU: :"PUNKTE.
...:";Z*RU: :
1970 GES=GES+(Z*RU)
1980 IF GES<HIGH THEN 2000
1990 HIGH=GES
2000 PRINT "GESAMT....:";GES
: :"HIGHSCORE :";HIGH: : :"W
EITER MIT FEUERKNOPF": :"AUF
HOEHREN MIT <A>"
2010 ZX=0
2020 CALL KEY(1,K,S)
2030 IF S=0 THEN 2020
2040 IF K=1 THEN 2350
2050 CALL SOUND(100,440,0)
2060 FOR I=1 TO 24
2070 FOR I1=1 TO 32
2080 A(I,I1)=0
2090 NEXT I1
2100 NEXT I
2110 GES=0
2120 GOTO 570
2130 REM GESCHAFFT !
2140 CALL SOUND(200,262,3,33
0,2,392,1)
2150 CALL SOUND(200,247,3,34
9,2,523,1)
2160 CALL SOUND(200,220,3,39
2,2,392,1)
2170 CALL SOUND(200,220,3,44
0,2,523,1)
2180 CALL SOUND(400,196,3,49
4,2,659,1)
2190 CALL CLEAR
2200 PRINT TAB(5);"BRAVO !":
 :"SIE HABEN DIE";RU;". RUND
E": :"GESCHAFFT.": : :
2210 PRINT "PILZE.....:";Z:
:"RUNDE.....:";RU: :"BONUS..
...:";5*RU: :"PUNKTE....:";Z
*RU+(5*RU): :
2220 GES=Z*RU+(5*RU)+GES
2230 PRINT "GESAMT....:";GES
: :"HIGHSCORE :";HIGH: : :"W
EITER MIT FEUERKNOPF"
2240 RU=RU+1
2250 CALL KEY(1,K,S)
2260 IF S=0 THEN 2250
2270 CALL SOUND(100,440,0)
2280 FOR I=1 TO 24
2290 FOR I1=1 TO 32
2300 A(I,I1)=0
2310 NEXT I1
2320 NEXT I
2330 Z=0
2340 GOTO 880
2350 END
```

LISTINGS

Fortsetzung von Seite 39

LERN-PARTNER

1. VORMENÜ

Nach dem Titelbild erscheint ein kleines Vormenü. Wurde bereits eine Textdatei auf Kassette erstellt, deren Stoff nun verarbeitet werden soll, ist die Funktion 2 (DATEN VON CS1 LADEN) zu wählen, andernfalls Funktion 1 (NEUEN TEXT EINGEBEN).

1.1 Neuen Text eingeben

Zunächst erscheint ein Bild, das die Eingabe eines Titels, eines Autors oder eines Kennworts verlangt. Damit wird die Datei spezifiziert.
Danach erscheint die Eingabemaske für die Datensätze. Das Eingabefeld besteht aus sechs Zeilen. Der Text ist zeilenweise einzugeben, jede Zeile mit Enter abzuschließen. Die Eingabe in einer Maske wird beendet, wenn die sechs Zeilen voll sind, oder eine Leerzeile an das Programm übergeben wird. Das Programm wechselt in den Befehlsmodus.
Mit ENTER erscheint die nächste Eingabemaske, mit BACK gelangt man zurück ins Menü, mit PROC'D kann der momentan gezeigte Datensatz korrigiert werden. Diese Option entspricht der Option 3 des Hauptmenüs (TEXT ERWEITERN).

1.2. Daten von CS1 laden

Eine Kassettendatei kann durch diese Option zur Weiterverarbeitung geladen werden. Ist eine Datei gefunden, wird der Titel (Autor/Kennwort) auf dem Bildschirm ausgegeben. Das Programm erwartet nun eine Angabe, ob es sich um die ge-

wünschte Datei handelt. Wenn ja, wird sie eingelesen, wobei jeder Datensatz auf dem Bildschirm erscheint. Andernfalls gelangt man ins Vormenü.

2. HAUPTMENÜ

Das Hauptmenü bietet acht weitere Optionen:

2.1 Text lernen

Datensatz für Datensatz zeigt das Programm in der oberen Hälfte der Bildschirmmaske ein Textstück an und wartet, bis Sie glauben, es gelernt zu haben. Drücken Sie nun eine Taste, und der Text verschwindet. Nun müssen Sie Zeile für Zeile den Text auswendig eingeben. Es werden nur soviele Zeilen angenommen, wie sie auch der Originaltext hat.
Durch ENTER wird das Programm mit der Lernkontrolle beginnen.
Dabei wird jede Abweichung vom Original als Fehler gewertet (Rechtschreibprüfung). Jede Schreibstelle, die vom Original abweicht, wird angezeigt. Der übrige Originaltext bleibt unsichtbar.
Haben Sie eine bestimmte Fehlerzahl im Verhältnis zur Datensatzlänge überschritten, werden Sie aufgefordert, einen neuen Versuch anzugehen.
Haben Sie in einem zweiten Durchgang die Fehlergrenze wieder überschritten, müssen Sie den Datensatz neu lernen.
Die Lektion ist beendet, wenn Sie alle Datensätze durchgearbeitet haben. Vorzeitig können Sie durch BACK ins Menü zurück, wenn sich der Computer im Befehlsmodus befindet.
Diese Lektion ist für Rechtschreibübungen geeignet.

2.2 Text abfragen

Lektion 2 arbeitet prinzipiell so, wie Lektion 1. Besonderheiten:

- Text wird nicht vorgestellt, sondern ist sofort auswendig einzugeben.
- Die Fehler werden nur gezählt, nicht sichtbar korrigiert.
- Im Befehlsmodus ist die Taste AID hinzugekommen. Mit ihr kann der Originaltext in der oberen Bildschirmhälfte sichtbar gemacht werden.

2.3 Text erweitern

Siehe oben NEUEN TEXT EINGEBEN.

2.4 Text korrigieren

Muß das Original nachträglich korrigiert werden, geben Sie die entsprechende Nummer des Datensatzes an. Der Datensatz erscheint in der Maske und kann nun überschrieben werden.
ACHTUNG: Eine eingeschobene Leerzeile im Datensatz bewirkt, daß die restlichen Zeilen nicht erkannt werden. Beispiel: Ein Datensatz hat 6 Zeilen. Löscht man Zeile 4, sind auch die Zeilen 5 und 6 gelöscht.

2.5 Text auflisten

Diese Option zeigt die gesamte Datei Datensatz für Datensatz. Die Anzeige stoppt nach jedem Datensatz, um Ihre Anweisungen abzuwarten: Mit der Anzeige fortfahren oder zurück ins Menü.

2.6 Text in CS1 speichern

Mit dieser Routine kann die Datei gespeichert werden. Jeder Datensatz wird auf dem Bildschirm angezeigt.

2.7 Ende

Das Programm verlangt zusätzlich eine Wahlbestätigung, was Datenverlust vorbeugt.

2.8 Erase, Speicher löschen

Will man mehrere Texte oder Teile eines zu umfangreichen Textes hinter-

Bitte lesen Sie weiter auf Seite 46

```
10 ! **********************
11 ! *                    *
12 ! *    LERNPARTNER     *
13 ! *                    *
14 ! *    Copyright by    *
15 ! *                    *
16 ! *  Reiner  Makohl    *
17 ! *                    *
19 ! * Benoetigte Geraete *
20 ! *  TI99/4A  Konsole  *
21 ! *     Ext. Basic     *
22 ! *    Cassettenrec.   *
23 ! *                    *
26 ! *  Speicherbelegung  *
27 ! *     5645 Bytes     *
28 ! *                    *
29 ! **********************
100 ! *  INITIALISIERUNG   *
110 ON WARNING NEXT :: DIM V
$(254):: K$="LERNPARTNER"
120 A$="00440038447C44440044
007C4444447C0044004444444438
" :: CALL CHAR(91,A$,123,A$&
"000000384448444C",136,"3C42
99A1A199423C")
130 CALL CHAR(128,"3E4151511
12E24C20201010102020C0000000
000446AAA9282028261",132,"00
FF",133,RPT$("01",8)&RPT$("8
0",8)&RPT$("0",12)&"FF")
140 CALL CLEAR :: CALL SCREE
N(5):: FOR I=0 TO 14 :: CALL
 COLOR(I,2,15):: NEXT I :: C
ALL MAGNIFY(3):: CALL SPRITE
(#1,128,2,33,117)
150 CALL HCHAR(10,9,30,15)::
 DISPLAY AT(12,9):K$ :: CALL
 HCHAR(14,9,30,15):: DISPLAY
 AT(22,3):CHR$(136)&"1984 by
  REINER MAKOHL"
160 I=I+1 :: GOSUB 710 :: IF
 K<13 AND I<250 THEN 160
170 ! *       VORMENUE     *
180 ! *  ART DER EINGABEN   *
190 CALL COLOR(#1,1):: CALL
D("14EINGABE"):: DISPLAY AT(
8,1):"DRJCKE": : :"1 > NEUEN
 TEXT EINGEBEN": :"2 > TEXT
VON CS1 LADEN"
200 GOSUB 710 :: IF K=49 THE
N 450 ELSE IF K<>50 THEN 200
210 ! *    TEXT VON CS1    *
220 CALL CLEAR :: PRINT "* D
ATEI LADEN";TAB(26);"CS1": :
 :: ON ERROR 1050
230 OPEN #1:"CS1",INPUT ,INT
ERNAL,FIXED 192 :: INPUT #1:
A$ :: IF A$<>K$ THEN PRINT :
 :"* ERROR - FALSCHE DATEI!"
 :: CLOSE #1 :: GOTO 190
240 INPUT #1:V$(0),Q :: PRIN
T : :V$(0): :"* DATEI RICHTI
G? (J/N)": :
```

```
250 GOSUB 710 :: IF K=78 THE
N Q=0 :: CALL HCHAR(22,28,78
):: CLOSE #1 :: GOTO 190 ELS
E IF K=74 THEN CALL HCHAR(22
,28,74)ELSE 250
260 CALL M(LEN(V$(0)),24,G):
: FOR J=1 TO Q :: INPUT #1:V
$(J):: PRINT V$(J): : :: CAL
L M(LEN(V$(J)),24,G):: NEXT
J :: CLOSE #1 :: Q=Q+1
270 ! *        MENUE         *
280 CALL D("05 WAHLLISTE")
290 DISPLAY AT(4,1):"DRJCKE"
: : :"1 > TEXT LERNEN": :"2
> TEXT ABFRAGEN": :"3 > TEXT
 ERWEITERN": :"4 > TEXT KORR
IGIEREN"
300 DISPLAY AT(15,1):"5 > TE
XT AUFLISTEN": :"6 > TEXT IN
 CS1 SPEICHERN": :"7 > ENDE"
: :"<ERASE> SPEICHER L\SCHEN
":TAB(9);"NEUEINGABE"
310 GOSUB 710 :: IF K=7 THEN
 660 ELSE IF K<49 OR K>55 TH
EN 310 ELSE ON K-48 GOTO 330
,400,450,520,560,590,630
320 ! *   1. TEXT LERNEN   *
330 CALL D("04TEXT..LERNEN")
:: Y=1 :: FOR J=1 TO Q-1
340 GOSUB 890 :: DISPLAY AT(
3,1):"LERNE JETZT BITTE DS#"
;J :: DISPLAY AT(5,1):V$(J):
: DISPLAY AT(18,2):"FERTIG?
DRJCKE EINE TASTE!"
350 GOSUB 710 :: IF K<13 THE
N 350
360 GOSUB 890 :: DISPLAY AT(
13,1):"KONTROLLE!":"GIB NUN
BITTE DS#";J;"EIN:" :: GOSUB
 990
370 IF F=2 AND W=1 THEN GOSU
B 730 :: ON K GOTO 340,680 E
LSE ON F+1 GOSUB 760,750,740
 :: IF K=2 THEN 680 ELSE IF
F=2 THEN W=1 :: GOTO 360
380 NEXT J :: GOTO 680
390 ! *  2. TEXT ABFRAGEN  *
400 CALL D("13TEXT  ABFRAGEN
"):: Y=2 :: FOR J=1 TO Q-1
410 GOSUB 890 :: DISPLAY AT(
13,1):"GIB NUN DS#";J;"EIN!"
:"HINWEIS:";INT(LEN(V$(J))/2
9)+1;"- ZEILER:"
420 GOSUB 990 :: ON F+1 GOSU
B 760,750,740 :: IF K=2 THEN
 680 ELSE IF F=2 THEN 410
430 NEXT J :: GOTO 680
440 ! *  3. TEXT  EINGABE  *
450 CALL D("16TEXTEINGABE"):
: GOSUB 910 :: IF Q>0 THEN 4
70 ELSE DISPLAY AT(8,1):"TIT
EL/AUTOR/KENNWORT" :: J=0 ::
 GOSUB 940
460 GOSUB 870 :: IF K=.5 THE
N 450 ELSE IF K=2 THEN 680
470 FOR I=0 TO 2 STEP 2 :: F
OR J=11 TO 16 :: CALL HCHAR(
J,3,30+I,28):: NEXT J :: NEX
T I
480 DISPLAY AT(8,1):"DATENSA
TZ #";Q :: CALL M(0,20,G)::
IF G=7056 THEN GOSUB 790 ::
ON K GOTO 480,680 ELSE J=Q :
: GOSUB 940
490 IF V$(Q)="" THEN GOSUB 8
00 ELSE GOSUB 870 :: IF K=.5
 THEN 480
500 ON K GOTO 470,680
510 ! * 4. TEXT-KORREKTUR *
520 CALL D("09TEXT KORRIGIER
EN"):: GOSUB 910 :: DISPLAY
AT(8,1):"DATENSATZ?";TAB(25)
;Q-1
530 ACCEPT AT(8,12)SIZE(3)VA
LIDATE(DIGIT)BEEP:J :: IF J>
Q-1 THEN 530 ELSE DISPLAY AT
(11,1):V$(J):: CALL M(-LEN(V
$(J)),20,G):: V$(J)=""
540 GOSUB 940 :: IF V$(J)=""
 THEN 540 ELSE CALL HCHAR(8,
18,32,11):: GOSUB 800 :: ON
K GOTO 520,680
550 ! * 5. TEXT AUFLISTEN *
560 CALL CLEAR :: FOR J=0 TO
 Q-1 :: PRINT "DS#";J: :V$(J
): : :: GOSUB 800 :: IF K=2
THEN 680
570 NEXT J :: GOTO 680
580 ! * 6. TEXT SPEICHERN *
590 CALL CLEAR :: PRINT "* T
EXT SPEICHERN.........CS1":
: :: ON ERROR 1050
600 OPEN #1:"CS1",OUTPUT,INT
ERNAL,FIXED 192 :: PRINT #1:
K$ :: PRINT : :"DATEI: ";K$;
Q-1;"DS": :V$(0): : :: PRINT
 #1:V$(0),Q-1
610 FOR J=1 TO Q-1 :: PRINT
V$(J): : :: PRINT #1:V$(J)::
 NEXT J :: CLOSE #1 :: GOTO
280
620 ! * 7. PROGRAMM-ENDE  *
630 CALL D("07ENDE"):: DISPL
AY AT(10,8):"TEXT GELERNT?":
 :TAB(6);"TEXT GESPEICHERT?"
: :TAB(9);"NEIN > BACK": :TA
B(9);"JA...> QUIT"
640 CALL COLOR(#1,2):: GOSUB
 710 :: IF K=15 THEN CALL CO
LOR(#1,1):: GOTO 280 ELSE 64
0
650 ! * 8. DATEN LOESCHEN *
660 CALL HCHAR(21,2,42):: RU
N
670 ! *  RETURN TO MENUE  *
680 CALL CLEAR :: DISPLAY AT
```

```
(12,6):"DURCHGANG BEENDET!"
:: W=0 :: FOR I=1 TO 400 ::
NEXT I :: GOTO 280
690 ! *  UNTERPROGRAMME   *
700 ! 1. CALL KEY
710 CALL KEY(0,K,K):: RETURN
720 ! 2. BEWERTUNGSTEXTE
730 A$="TEXT NEU LERNEN!" ::
 W=0 :: GOTO 770
740 A$=" NEUER VERSUCH!" ::
GOTO 770
750 A$="...NOCH GUT!" :: GOT
O 770
760 A$="...SEHR GUT!"
770 DISPLAY AT(23,13)BEEP:A$
 :: ON Y GOTO 800,810
780 ! 3. ORDER-INPUT
790 A$="END >BACK" :: X=1 ::
 GOTO 830
800 A$="GO >ENTER..........E
ND >BACK" :: X=1 :: GOTO 830
810 A$="GO>ENTER..END>BACK..
VIEW>AID" :: X=2 :: GOTO 830
820 A$="GO>ENTER END>BACK KO
RR>PROCD" :: X=3
830 DISPLAY AT(24,1):A$
840 CALL KEY(0,K,S):: IF S<1
 THEN 840 ELSE IF K=13 OR K=
15 OR(X=3 AND K=12)THEN K=(K
-11)/2 :: CALL HCHAR(24,3,32
,28):: RETURN
850 IF K=1 AND X=2 THEN DISP
LAY AT(5,1):V$(J):: GOTO 840
 ELSE 840
860 ! 4. ORDERANFORDERUNG
870 GOSUB 820 :: IF K=.5 THE
N CALL M(-LEN(V$(Q)),20,G)::
 V$(Q)="" :: RETURN ELSE Q=Q
+1 :: RETURN
880 ! 5. BILDSCHIRMMASKE 1
890 CALL HCHAR(3,1,32,704)::
 FOR I=4 TO 15 STEP 11 :: GO
SUB 920 :: NEXT I :: FOR I=5
 TO 10 :: CALL HCHAR(I,3,30,
28):: NEXT I :: RETURN
900 ! 6. BILDSCHIRMMASKE 2
910 I=10
920 CALL HCHAR(I,3,135,28)::
 CALL HCHAR(I+7,3,132,28)::
CALL VCHAR(I+1,2,133,6):: CA
LL VCHAR(I+1,31,134,6):: RET
URN
930 ! 7. EINGABEN WAHL 3/4
940 FOR I=11 TO 16 :: ACCEPT
 AT(I,1)SIZE(-28):A$ :: IF A
$="" THEN 960 ELSE V$(J)=V$(
J)&A$&RPT$(" ",28-LEN(A$))
950 NEXT I
960 IF G+LEN(V$(Q))>7056 THE
N CALL SOUND(220,220,0):: DI
SPLAY AT(8,16):"TEXT ZU LANG
!" :: V$(J)="" ELSE CALL M(L
EN(V$(J)),20,G)
```

LISTINGS

```
970 RETURN
980 ! 8. EINGABEN WAHL 1/2
990 F=0 :: L=1 :: FOR I=1 TO
 INT(LEN(V$(J))/29)+1 :: T$=
SEG$(V$(J),L,28):: ACCEPT AT
(15+I,1):A$ :: IF (A$&RPT$("
 ",28-LEN(A$)))=T$ THEN 1030
1000 FOR K=1 TO 28 :: CALL G
CHAR(15+I,K+2,C):: IF C=ASC(
SEG$(T$,K,1))THEN 1020 ELSE
F=F+1 :: DISPLAY AT(23,1):F;
"FEHLER!"
1010 IF Y=1 THEN CALL SOUND(
-2,999,3):: DISPLAY AT(4+I,K
)SIZE(1):SEG$(T$,K,1)
1020 NEXT K
1030 L=L+28 :: NEXT I :: IF
F>INT(LEN(V$(J))/56)+1 THEN
F=2 :: RETURN ELSE IF F>0 TH
EN F=1 :: RETURN
1040 ! 9. ON ERROR CS1
1050 CALL ERR(FC,FT,FG,FZ)::
 IF FC=130 THEN PRINT : :"*
I/O ERROR 26" ELSE IF FC=109
 THEN PRINT : :"* FILE ERROR
" ELSE RETURN
1060 CLOSE #1 :: IF FZ<300 T
HEN Q=0 :: V$(0)="" :: RETUR
N 190 ELSE RETURN 280
1070 ! 10. KOPFZEILE
1080 SUB D(D$):: CALL CLEAR
:: CALL SCREEN(VAL(SEG$(D$,1
,2)):: DISPLAY AT(1,(32-LEN
(D$))/2):SEG$(D$,3,28):RPT$(
CHR$(132),28):: SUBEND
1090 ! 11. SIZE-CONTRL.
1100 SUB M(M,X,G):: G=G+M ::
 IF G>7055 THEN G=7056 :: CA
LL SOUND(220,220,3):: DISPLA
Y AT(X,1):"* MEMORY FULL *"
:: SUBEXIT
1110 DISPLAY AT(X,1):USING "
#### BYTES FREE":7056-G :: S
UBEND
```



LERNHILFE

einander bearbeiten, muß zwischendurch der Speicherinhalt im Computer gelöscht werden. Dies geschieht mit der Taste ERASE. Wird sie betätigt, sind alle Daten verloren. Bis die Hauptwahlliste verschwindet, und das Programm neu startet, dauert es einen Moment.

3. SPEICHERPLATZ UND TEXTLÄNGEN

Das Programm kontrolliert ständig den noch freien Speicherplatz und gibt ihn auf dem Bildschirm aus. Lange Texte, die den Speicherplatz überschreiten würden, müssen gesplittet werden: Text teilweise bearbeiten, danach auf Kassette abspeichern, den Speicherinhalt löschen und mit dem Text fortfahren.
Ist der Speicher voll, kommt es zur Meldung "*MEMORY FULL*". Eingaben werden dann nicht mehr zugelassen. Textkorrekturen können nur vorgenommen werden, wenn sich dadurch nicht die Länge eines Datensatzes vergrößert. Sollte das dennoch nötig sein, muß man den letzten Datensatz zunächst kürzen oder löschen.
Würde durch einen Datensatz oder eine Korrektur der Speicherplatz überschritten, erscheint die Meldung "TEXT ZU LANG, NEUEINGABE!", und der Datensatz wird nicht akzeptiert.

4. ZUM LISTING

Um das Listing übersichtlich zu gestalten, wurde es stark mit REM-Zeilen durchsetzt. Das Programm läuft durch die Speicherplatzkontrollen allerdings nur ohne diese REM-Zeilen einwandfrei. Die REM-Zeilen sind also für das lauffähige Programm zu entfernen!
Der Befehl CALL KEY (0,K,K) ist kein Fehler! Hier wird eine zusätzliche Variable eingespart.

GOLDSUCHER

Goldsucher ist ein Strategiespiel für einen Spieler. Durch zehn Maschinen-Unterprogramme für das MINIMEM-Modul, schnellen Bildaufbau und 32 Spielvarianten reicht es qualitativ trotz TI-(MINIMEM)-BASIC beinahe an TI-Spielmodule heran.
Aufgabe des Spielers ist es, mit möglichst wenigen Spielzügen im Spielfeld unsichtbar verteilte Goldstücke zu finden. Dabei führen Strategien und Kombinationsgabe zu den besten Erfolgen.

HANDHABUNG DER PROGRAMME I UND II

Mit Goldsucher I werden die notwendigen Daten und Maschinen-Programme in das MINIMEM-Modul geladen: MINIMEM-Modul einschieben, Goldsucher I laden und starten.
Da der gesamte Ladevorgang recht langwierig ist, sollte man danach direkt den MINIMEM-RAM-Bereich >7000 bis >7FFF speichern und zum erneuten Laden benutzen. Eine Operationsanweisung wird am Ende des Programms gegeben.
Goldsucher II ist das eigentliche Spielprogramm und arbeitet nur mit geladenem MINIMEM-Modul.
Achtung: Bei Spielende reagiert der Computer genauso, als wäre die QUIT-Taste gedrückt worden. Vor dem ersten Spielstart sollte also das Programm schon auf Band gespeichert sein!

SPIELREGELN, SPIELABLAUF

Die Titelgrafik erklärt als Kurzspielanleitung alle Spielelemente und Symbole.
In der Menütafel werden die Spieloptionen vom Spieler bestimmt:
- einfaches oder komplexes Spielfeld
- 3,4,5 oder 6 zu suchende Goldstücke
- mit oder ohne Detektorstörfeld
- mit oder ohne Fallgrube

Das Spielfeld besteht aus 6x9 Punkten, die durch Wege miteinander verbunden sind. Im einfachen Spielfeld sind nur waagerechte und senkrechte Wege vorhanden, im komplexen Spielfeld kommen noch diagonale Wege hinzu.
Die Spielfigur kann mit Hilfe des Joysticks über die Wege jeden Punkt des Spielfeldes erreichen und startet immer in ihrem Heimfeld. Läßt man die Figur auf einem Punkt stehen und wird dann die Aktionstaste des Joysticks betätigt, ist der Golddetektor aktiviert.
Der Golddetektor meldet, ob sich im abgefragten Punkt ein Goldstück befindet. Wenn ja, erscheint im betreffenden Punkt das Symbol des Goldbarren, andernfalls wird ein Zahlenwert angezeigt. Dieser Zahlenwert ist die Summe aller Goldstücke, die auf allen Wegen liegen, die sich im betreffenden Punkt kreuzen!
Erläuterung: Im einfachen Spielfeld kreuzen sich in jedem Punkt ein senkrechter und ein waagerechter Weg. Befindet sich im abgefragten Punkt kein Goldstück, wird angezeigt, wieviele Goldstücke in den übrigen 14 Punkten der beiden Wege verborgen sind. Dabei werden bereits entdeckte Goldstücke mitgezählt.
Anhand mehrerer Zahlenwerte lassen sich nun Punkte ermitteln, die mit Sicherheit ein Goldstück verbergen. Eine gute Kombinationsgabe und eine sinnvolle Strategie führen in wenigen Zügen zum Erfolg!
Im komplexen Spielfeld werden bei den Zahlenwerten auch noch die diagonalen Wege berücksichtigt, was den Schwierigkeitsgrad merklich steigert!
Das Detektorstörfeld ist eine tückische Sache: Zunächst unsichtbar wird irgendwo im Spielfeld ein Störfeld versteckt. In diesem Punkt befindet sich dann kein Goldstück. Das Störfeld beeinflußt alle Wege, die sich in ihm kreuzen. Die Zahlenwerte der betroffenen Punkte sind gegenüber ihrem tatsächlichen Wert um eins vermindert und erschweren dadurch die Ermittlungen. Trifft man allerdings auf das Störfeld selbst, erscheint sein Symbol, und man kann sich ausrechnen, welche Punkte falsche Werte anzeigen!
Die Fallgrube ist noch gemeiner, hier spielt nun auch das Glück seine Rolle: Wieder unsichtbar und irgendwo im Spielfeld wird die Fallgrube versteckt. Auch in diesem Punkt kann kein Goldstück liegen. Gerät man in die Fallgrube, ist man verloren, und das Spiel ist aus.
Jedoch gibt es noch die Fallenwarnungen: Auf dem waagerechten oder auf dem senkrechten Weg, der die Fallgrube kreuzt, werden Warnungen angebracht. Trifft man auf sie, erscheint ihr Symbol, und man kann sich ausrechnen, auf welchem Weg die Fallgrube liegt.
Allerdings können durch die Warnsymbole keine Detektorwerte angezeigt werden! Und: Auf dem Weg, der die Warnungen enthält, können trotzdem Goldstücke verborgen sein. Die Gefahr in die Falle zu geraten, ist groß!

Bitte lesen Sie weiter auf Seite 55

LISTINGS

```
10 REM********************
11 REM*   GOLDSUCHER I    *
12 REM*      (Loader)     *
13 REM*                   *
14 REM*  Copyright  by    *
15 REM*  Reiner Makohl    *
16 REM*                   *
17 REM*Benoetigte Geraete*
18 REM* TI99/4A  Konsole *
19 REM*     Minimemory    *
20 REM*     Joystick 1    *
21 REM*   Cassettenrec.   *
22 REM*                   *
23 REM* Speicherbelegung *
24 REM*    11193 Bytes    *
25 REM********************
26 REM
100 DATA 3E415151112E24C2,02
0101010202 0C
110 DATA 00000000446AAA92,82
028261
120 DATA 0000FFFFFFFF,000000
3C7EE7C3C
130 DATA 0000003C7EE7C3C3,00
000040C0C0C0C
140 DATA 0000007CFEC7C3C3,00
00003E7FE0C0E
150 DATA 00000042C3C3C3C3,00
00003E7FE0C0C
160 DATA 0000007EFFE0C0E,000
000FCFEC7C3C7
170 DATA CFCFC3C3C3E77E3C,C3
C3C3C3C3E77E3C
180 DATA C0C0C0C0C0E0FE7F,C3
C3C3C3C3C7FE7C
190 DATA 7C3E07030307FE7C,C0
C0C0C0C0E07F3E
200 DATA FFFFC3C3C3C3C342,FC
FCE0C0C0E0FF7E
210 DATA FEFCF0F8DCCEC7C3,0,
0,0,0,0,0
220 DATA 000101010101 01,102
828444483
230 DATA 00000000629508,0000
FFFFFFFF
240 DATA FFFFFFFFFFFFFFFF,FF
83452911294583
250 DATA 181818FFFF181818,0
260 DATA 0018242424 2418,0008
180808081C
270 DATA 0018240418203C,0018
2408042418
280 DATA 008018283C0808,003C
2038040438
290 DATA 0018203824 2418,DBDB
00DBDB00DBDB
300 DATA 183C7E7E7E7E3C18,00
000066
310 DATA 0010100000101,00402
000000402
320 DATA 0002040000204,080C0
EFFFF0E0C08,0
330 DATA 00070E1D3B0678,0000
0000107070F
340 DATA 0000000080E0E0F,F0E
0E08,0F070701
350 DATA 00000000FFFFFFFF,F0
F0F0F0F0F0F0F
360 DATA FFFFFFFF,0F0F0F0F0F
0F0F0F,0
370 DATA 30302478B038C88C,00
0028107C1028
380 DATA 0038440418044438,00
081828487C0808
390 DATA 007C407804044438,00
18204078444438
400 DATA 6,0210GOLDSUCHER..I
,0310===============
410 DATA 1003HILFSPROGRAMM Z
UM LADEN DES,1211MINI MEMORY
420 DATA 2321< 1984..[],2420
R.MAKOHL \^
430 REM TITELBILD
440 DATA 21,0204s`abcdefeghs
..s`abcdefeghs
450 DATA 0304sijklmjnopqs..s
ijklmjnopqs
460 DATA 0504sxsxsxsxsxsx..s
xsxsxsxsxsx
470 DATA 0604zyzyzyzyzyzy..z
yzyzyzyzyzy
480 DATA 0804JOYSTICK 1,0817
GOLD-DETEKTOR
490 DATA 0904FEUERTASTE,0917
WERT ABFRAGEN
500 DATA 1108SPIELER-,1122GO
LD-
510 DATA 1208FIGUR,1222BARRE
N
520 DATA 1408SPIELFELD,1422S
TOERFELD
530 DATA 1608WEGE,1622FALLE
540 DATA 1808ORTUNGS-,1822FA
LLEN-
550 DATA 1908WERTE,1922WARNU
NG
560 DATA 2104WEITER DURCH TA
STENDRUCK!
570 DATA 4,2,3,151,5,2,16,14
9,5
580 DATA 2,17,151,5,2,30,149
,5
590 DATA 18,1,3,148,28,7,3,1
50,28,1,3,144,1
600 DATA 1,16,145,1,1,17,144
,1,1,30,145,1
610 DATA 7,3,147,1,7,16,146,
1,7,17,147,1
620 DATA 7,30,146,1,11,4,153
,1,14,4,136,1
630 DATA 16,4,137,1,18,4,130
,1,11,18,143,1
640 DATA 14,18,125,1,16,18,1
35,1,18,18,126,1
```

```
650 DATA 8,15,9,15,11,6,14,6
,16,6,18,6
660 DATA 11,20,14,20,16,20,1
8,20
670 REM SPIELMENUE
680 DATA 15,0220SPIELFELD:,0
3201 EINFACH
690 DATA 04202 KOMPLEX,0720Z
AHL DER
700 DATA 0820GOLDSTUECKE:
710 DATA 0903SPIELOPTIONEN:.
..(3 BIS 6)
720 DATA 1103BITTE WAEHLEN
730 DATA 1203SIE AUS 32.....
..DETEKTOR-
740 DATA 1303MOEGLICHKEITEN.
..STOERFELD:
750 DATA 1403IHR SPIEL!.....
..0 OHNE
760 DATA 15201 MIT,1820FALLG
RUBE:
770 DATA 1905HIGHSCORE......
0 OHNE
780 DATA 20201 MIT,220700000
790 DATA 7,10,3,45,14,20,5,1
44,1,20,6,148,7
800 DATA 20,13,145,1,24,5,14
7,1,24,6,150,7
810 DATA 24,13,146,1
820 DATA 3,21,5,151,3,21,13,
149,3,3,18,141,2
830 REM FELDRAHMEN
840 DATA 7,0203SCORE...s`abc
defeghs...G=0:0
850 DATA 030300000 * sijklmj
nopqs,0426DST:
860 DATA 0503BONUS...sssssss
sssss
870 DATA 060300000...zzzzzzz
zzzzz...FGR:
880 DATA 2331[],2425 <1984\^
890 DATA 11,1,2,144,1,1,3,14
8,28,1,31,145,1
900 DATA 3,26,95,5,4,3,95,5,
4,11,123,12
910 DATA 4,23,124,3,5,26,95,
5,7,3,124,28
920 DATA 24,2,147,1,24,3,150
,22
930 DATA 13,2,2,151,22,2,31,
149,20
940 DATA 2,8,124,5,5,9,141,2
,2,10,124,5
950 DATA 2,23,124,5,2,25,124
,5,5,24,153,1
960 DATA 6,24,138,2,2,9,124,
1,4,9,124,1
970 DATA 2,24,124,1,3,24,42,
1
980 DATA 2,0101ssssssssssssx
sxsxsxsxsxs
990 DATA 0201zzzzzzzzzzzyzy
zyzyzyzyz
1000 REM TEXTE
1010 DATA 173,175,173,165,17
4,180,128,128
1020 DATA 162,169,180,180,16
5,129,167,128
1030 DATA 161,128,173,128,16
5,128,128,175
1040 DATA 128,182,128,165,12
8,178,174,165
1050 DATA 181,165,179,128,17
9,176,169,165
1060 DATA 172,159,128,136,17
0,143,174,137
1070 REM * DATA ZU TEIL 2 *
1080 REM  PROGRAMM
1090 DATA 1,1009PROGRAMM POK
EN
1100 DATA 192,59,192,123,192
,187,4,32
1110 DATA 96,36,5,128,6,2,22
,251
1120 DATA 4,91,192,123,192,5
9,192,187
1130 DATA 4,32,96,40,4,91,19
2,59
1140 DATA 192,123,192,187,19
2,251,4,32
1150 DATA 96,40,2,32,0,32,16
0,66
1160 DATA 6,3,22,249,4,91,4,
192
1170 DATA 2,1,0,1,4,32,96,68
1180 DATA 4,32,96,28,18,0,19
3,32
1190 DATA 131,74,4,91,194,13
9,6,160
1200 DATA 125,144,3,16,0,0,0
,16
1210 DATA 6,160,125,144,0,0,
128,0
1220 DATA 3,0,4,90,194,139,6
,160
1230 DATA 125,144,3,16,16,0,
0,8
1240 DATA 192,122,6,160,125,
164,3,24
1250 DATA 0,8,4,90,194,139,1
92,58
1260 DATA 6,160,125,176,115,
80,0,14
1270 DATA 0,7,4,90,194,75,6,
160
1280 DATA 125,220,6,160,126,
12,0,9
1290 DATA 6,160,125,174,0,1,
119,244
1300 DATA 0,9,0,7,6,160,125,
174
1310 DATA 0,22,120,51,0,9,0,
7
```

```
1320 DATA 6,160,125,144,0,20
2,220,0
1330 DATA 0,12,4,89,194,11,6
,160
1340 DATA 125,220,6,160,125,
162,113,48
1350 DATA 5,216,2,32,6,160,1
26,12
1360 DATA 0,2,6,160,126,12,0
,16
1370 DATA 6,160,125,162,115,
178,0,224
1380 DATA 2,32,6,160,125,244
,113,24
1390 DATA 4,88,194,11,6,160,
125,220
1400 DATA 6,160,126,12,0,2,6
,160
1410 DATA 125,174,0,49,117,2
10,0,14
1420 DATA 0,23,6,160,125,174
,1,2
1430 DATA 119,20,0,14,0,16,6
,160
1440 DATA 125,244,113,24,4,8
8,194,11
1450 DATA 6,160,125,220,6,16
0,126,12
1460 DATA 0,9,6,160,125,162,
125,0
1470 DATA 1,105,0,14,6,160,1
25,244
1480 DATA 113,24,4,88,194,11
,6,160
1490 DATA 126,28,6,160,125,1
62,120,114
1500 DATA 0,224,2,32,6,160,1
25,244
1510 DATA 113,32,4,88,194,11
,6,160
1520 DATA 126,28,6,160,125,1
62,122,146
1530 DATA 0,224,2,32,6,160,1
25,244
1540 DATA 113,40,4,88,194,11
,6,160
1550 DATA 125,198,5,129,2,3,
28,0
1560 DATA 2,2,124,178,196,13
1,4,32
1570 DATA 96,76,4,195,208,24
2,6,195
1580 DATA 2,5,96,0,192,4,208
,114
1590 DATA 176,69,4,32,96,36,
5,132
1600 DATA 6,3,22,248,4,88,19
4,11
1610 DATA 6,160,125,198,10,2
0,2,1
1620 DATA 124,208,160,68,6,1
60,125,164
```

LISTINGS

```
1630 DATA 0,138,0,12,2,0,0,1
70
1640 DATA 2,33,0,24,4,32,96,
40
1650 DATA 4,88,194,11,6,160,
125,144
1660 DATA 0,138,216,0,0,12,6
,160
1670 DATA 125,144,0,170,217,
0,0,12
1680 DATA 4,88,194,11,2,0,1,
68
1690 DATA 2,1,128,0,2,3,0,10
1700 DATA 2,2,0,24,6,160,125
,150
1710 DATA 2,32,0,8,6,3,22,24
8
1720 DATA 6,160,125,162,125,
14,1,136
1730 DATA 0,16,6,160,125,162
,125,30
1740 DATA 2,39,0,18,4,88,4,3
2
1750 DATA 96,24,0,32,4,91,0,
0
1760 REM NAMETABLE
1770 DATA 69,127,150,67,127,
98,71,127,74
1780 DATA 79,127,38,68,126,2
44,75,126,220
1790 DATA 78,126,196,82,126,
166,77,126,122
1800 DATA 84,126,76
1810 REM * TEIL 1 *
1820 REM * VDP-DATEN *
1830 CALL INIT
1840 CALL CLEAR
1850 CALL SCREEN(6)
1860 REM  COLOR
1870 CALL POKEV(792,175,175,
175,31,27,240,16,240)
1880 CALL LOAD(28952,175,175
,175,31,27,240,16,240)
1890 CALL LOAD(28960,175,175
,175,31,27,225,16,177)
1900 CALL LOAD(28968,175,175
,175,31,27,253,16,253)
1910 REM  CHARACTER
1920 CALL CHAR(60,"3C4299A1A
199423C")
1930 FOR I=91 TO 158
1940 READ A$
1950 CALL CHAR(I,A$)
1960 NEXT I
1970 GOSUB 2980
1980 J=68
1990 VDP=1496
2000 MM=28976
2010 GOSUB 3090
2020 REM  TITEL
2030 CALL HCHAR(1,1,32,704)
2040 GOSUB 2980
```

LISTINGS

```
2050 CALL HCHAR(4,4,123,26)
2060 GOSUB 3340
2070 GOSUB 3270
2080 FOR I=1 TO 10
2090 READ A,B
2100 CALL HCHAR(A,B,141)
2110 NEXT I
2120 A=2
2130 B=13
2140 C=7
2150 MM=29520
2160 GOSUB 3170
2170 J=68
2180 VDP=224
2190 MM=29618
2200 GOSUB 3090
2210 REM SPIELMENUE
2220 FOR I=1 TO 7
2230 CALL HCHAR(I,17,32,15)
2240 NEXT I
2250 CALL HCHAR(8,1,32,448)
2260 GOSUB 2980
2270 GOSUB 3270
2280 GOSUB 3340
2290 A=49
2300 B=13
2310 C=24
2320 MM=30162
2330 GOSUB 3170
2340 A=258
2350 MM=30484
2360 GOSUB 3170
2370 REM FELDRAHMEN
2380 CALL CLEAR
2390 GOSUB 2980
2400 GOSUB 3270
2410 GOSUB 3340
2420 A=1
2430 B=8
2440 C=7
2450 MM=30708
2460 GOSUB 3170
2470 A=22
2480 MM=30771
2490 GOSUB 3170
2500 REM FELD EINFACH
2510 FOR I=8 TO 20 STEP 3
2520 FOR J=3 TO 27 STEP 3
2530 CALL HCHAR(I,J,136)
2540 CALL HCHAR(I,J+1,137,2)
2550 CALL HCHAR(I,J+3,136)
2560 CALL VCHAR(I+1,J,138,2)
2570 CALL VCHAR(I+1,J+1,142,
2)
2580 CALL VCHAR(I+1,J+2,142,
2)
2590 CALL VCHAR(I+1,J+3,138,
2)
2600 CALL HCHAR(I+3,J,136)
2610 CALL HCHAR(I+3,J+1,137,
2)
2620 CALL HCHAR(I+3,J+3,136)
2630 NEXT J
2640 NEXT I
2650 J=68
2660 VDP=224
2670 MM=30834
2680 GOSUB 3090
2690 REM FELD KOMPLEX
2700 FOR I=9 TO 21 STEP 3
2710 FOR A=4 TO 28 STEP 3
2720 CALL HCHAR(I,A,139)
2730 CALL HCHAR(I,A+1,140)
2740 CALL HCHAR(I+1,A,140)
2750 CALL HCHAR(I+1,A+1,139)
2760 NEXT A
2770 NEXT I
2780 MM=31378
2790 GOSUB 3090
2800 REM OSZILLOGRAMM
2810 CALL CLEAR
2820 GOSUB 2980
2830 FOR I=0 TO 23
2840 CALL PEEKV(0+I,A)
2850 CALL PEEKV(32+I,B)
2860 CALL LOAD(31952+I,A)
2870 CALL LOAD(31976+I,B)
2880 NEXT I
2890 REM TEXTE UND HS
2900 FOR I=0 TO 47
2910 READ A
2920 CALL LOAD(32000+I,A)
2930 NEXT I
2940 CALL LOAD(32048,48,48,4
8,48,48,48)
2950 GOTO 3430
2960 REM * UNTERPROGRAMME *
2970 REM  PRINT
2980 READ D
2990 FOR I=1 TO D
3000 READ A$
3010 A=VAL(SEG$(A$,1,2))
3020 B=VAL(SEG$(A$,3,2))
3030 FOR C=1 TO LEN(A$)-4
3040 CALL HCHAR(A,B+C-1,ASC(
SEG$(A$,C+4,1)))
3050 NEXT C
3060 NEXT I
3070 RETURN
3080 REM PEEK/POKE 1
3090 X=0
3100 FOR I=1 TO J
3110 CALL PEEKV(VDP+X,A,B,C,
D,E,F,G,H)
3120 CALL LOAD(MM+X,A,B,C,D,
E,F,G,H)
3130 X=X+8
3140 NEXT I
3150 RETURN
3160 REM PEEK/POKE 2
3170 X=0
3180 FOR I=A TO (C-1)*32+A S
TEP 32
```

```
3190 FOR J=0 TO B
3200 CALL PEEKV(I+J,D)
3210 CALL LOAD(MM+X+J,D)
3220 NEXT J
3230 X=X+B+1
3240 NEXT I
3250 RETURN
3260 REM  HCHAR
3270 READ A
3280 FOR I=1 TO A
3290 READ B,C,D,E
3300 CALL HCHAR(B,C,D,E)
3310 NEXT I
3320 RETURN
3330 REM VCHAR
3340 READ A
3350 FOR I=1 TO A
3360 READ B,C,D,E
3370 CALL VCHAR(B,C,D,E)
3380 NEXT I
3390 RETURN
3400 REM * TEIL 2 *
3410 REM *TMS9900 PROGRAMM*
3420 REM PROGRAMME
3430 CALL CLEAR
3440 GOSUB 2980
3450 X=0
3460 FOR I=1 TO 66
3470 READ A,B,C,D,E,F,G,H
3480 CALL LOAD(32144+X,A,B,C
,D,E,F,G,H)
3490 X=X+8
3500 NEXT I
3510 REM LINK-NAME
3520 X=0
3530 FOR I=1 TO 10
3540 READ A,B,C
3550 CALL LOAD(32688+X,A,32,
32,32,32,32,B,C)
3560 X=X+8
3570 NEXT I
3580 REM FFAM+LFAM
3590 CALL LOAD(28700,127,158
,127,176)
3600 REM * ENDE *
3610 PRINT "* IST DIESE PROG
RAMM AUF......DATENTRAEGER G
ESPEICHERT?": :"..WENN JA, D
ANN": :
3620 PRINT "* DRUECKE QUIT":
 :"* WAEHLE EASY BUG": :"* S
AVE >7000 TO 7FFF": :"* LADE
 DAS MINI MEMORY BEI"
3630 PRINT "..BEDARF MIT DEM
 MASCHINEN-...PROGRAMM PER L
(EASY BUG)": :"* LADE ZUM SP
IELEN DAS"
3640 PRINT "..PROGRAMM GOLDS
UCHER II": :
```

LISTINGS

```
10 REM*******************
11 REM*   GOLDSUCHER II   *
12 REM*      (Teil 2)     *
13 REM*                   *
14 REM*  Copyright  by    *
15 REM*  Reiner Makohl    *
16 REM*                   *
17 REM*Benoetigte Geraete*
18 REM* TI99/4A  Konsole *
19 REM*     Minimemory    *
20 REM*     Joystick 1    *
21 REM*    Cassettenrec.  *
22 REM*                   *
23 REM* Speicherbelegung *
24 REM*      7465 Bytes   *
25 REM*******************
26 REM
100 CALL SCREEN(14)
110 CALL CLEAR
120 CALL CHAR(60,"3C4299A1A1
99423C")
130 CALL CHAR(159,"F")
140 CALL LINK("T")
150 DIM W(17,5),S(17,5),WA(5
),SE(9),D1(14),D2(14)
160 DEF P$(X)=SEG$(".....",1
,5-LEN(STR$(X)))&STR$(X)
170 CALL PEEK(32048,SE(1),SE
(2),SE(3),SE(4),SE(5))
180 FOR I=1 TO 5
190 H$=H$&CHR$(SE(I))
200 NEXT I
210 HS=VAL(H$)
220 X=13
230 Y=255
240 GOSUB 2990
250 CALL LINK("M")
260 CALL SCREEN(6)
270 CALL LINK("D",678,H$)
280 X=49
290 Y=50
300 GOSUB 2990
310 SP=K-48
320 CALL POKEV(81,128,"",113
,128,"",49+SP*32,250,"",273,
237)
330 X=51
340 Y=54
350 GOSUB 2990
360 GL=K-48
370 CALL POKEV(273,248+GL,""
,433,237,"",465,237)
380 X=48
390 Y=49
400 GOSUB 2990
410 DS=K-48
420 CALL POKEV(433,128,"",46
5,128,"",433+DS*32,250)
430 CALL POKEV(593,237,"",62
5,237)
440 GOSUB 2990
```

```
450 FG=K-48
460 CALL POKEV(593,128,"",62
5,128,"",593+FG*32,250)
470 GOSUB 2960
480 CALL LINK("R")
490 CALL SCREEN(11)
500 RANDOMIZE
510 FOR GOLD=1 TO GL-1
520 GOSUB 2930
530 W(X,Y)=1
540 NEXT GOLD
550 GOLD=0
560 GOSUB 2930
570 W(X,Y)=1
580 FOR Y=0 TO 5
590 FOR X=4 TO 13
600 IF W(X,Y)=0 THEN 630
610 GOLD=GOLD+1
620 GOSUB 3310
630 NEXT X
640 NEXT Y
650 IF GOLD<GL THEN 550
660 IF DS=0 THEN 700
670 GOSUB 2930
680 IF W(X,Y)<>0 THEN 670
690 W(X,Y)=-1
700 FOR Y=0 TO 5
710 WA(Y)=W(4,Y)+W(5,Y)+W(6,
Y)+W(7,Y)+W(8,Y)+W(9,Y)+W(10
,Y)
720 WA(Y)=WA(Y)+W(11,Y)+W(12
,Y)+W(13,Y)
730 GOSUB 3310
740 NEXT Y
750 FOR X=4 TO 13
760 SE(X-4)=W(X,0)+W(X,1)+W(
X,2)+W(X,3)+W(X,4)+W(X,5)
770 GOSUB 3310
780 NEXT X
790 IF SP=1 THEN 880
800 FOR X=0 TO 12
810 D1(X+1)=W(X,0)+W(X+1,1)+
W(X+2,2)+W(X+3,3)+W(X+4,4)+W
(X+5,5)
820 GOSUB 3310
830 NEXT X
840 FOR X=5 TO 17
850 D2(X-4)=W(X,0)+W(X-1,1)+
W(X-2,2)+W(X-3,3)+W(X-4,4)+W
(X-5,5)
860 GOSUB 3310
870 NEXT X
880 FOR X=4 TO 13
890 FOR Y=0 TO 5
900 S(X,Y)=WA(Y)+SE(X-4)+D1(
X-Y+1)+D2(X+Y-4)
910 GOSUB 3310
920 NEXT Y
930 NEXT X
940 FOR Y=0 TO 5
950 FOR X=4 TO 13
```

LISTINGS

```
960 IF W(X,Y)<>-1 THEN 980
970 S(X,Y)=-4
980 IF W(X,Y)<1 THEN 1000
990 S(X,Y)=-1
1000 GOSUB 3310
1010 NEXT X
1020 NEXT Y
1030 IF FG=0 THEN 1200
1040 GOSUB 2930
1050 IF W(X,Y)<>0 THEN 1040
1060 S(X,Y)=-2
1070 W(X,Y)=-2
1080 IF SP=2 THEN 1150
1090 FOR Y=0 TO 5
1100 IF W(X,Y)<>0 THEN 1120
1110 S(X,Y)=-3
1120 GOSUB 3310
1130 NEXT Y
1140 GOTO 1200
1150 FOR X=4 TO 13
1160 IF W(X,Y)<>0 THEN 1180
1170 S(X,Y)=-3
1180 GOSUB 3310
1190 NEXT X
1200 BO=SP*10000+GL*5000+DS*
20000+FG*20000+9000
1210 FA=(BO-9000)/50
1220 IF SP=2 THEN 1290
1230 CALL LINK("N")
1240 CALL SCREEN(6)
1250 CALL POKEV(785,17)
1260 C1=177
1270 C2=97
1280 GOTO 1340
1290 CALL LINK("K")
1300 CALL SCREEN(12)
1310 CALL POKEV(785,17)
1320 C1=253
1330 C2=29
1340 CALL POKEV(61,144+GL,""
,125,174-4*DS,"",189,174-4*F
G)
1350 CALL LINK("D",162,P$(BO
))
1360 FOR I=1 TO 3
1370 GOSUB 3340
1380 NEXT I
1390 FOR GOLD=1 TO GL
1400 FOR I=1 TO 3
1410 CALL JOYST(1,X,Y)
1420 IF (X=0)+(Y=-4)=-2 THEN
 1510
1430 NEXT I
1440 CALL POKEV(799,C2)
1450 FOR I=1 TO 3
1460 CALL JOYST(1,X,Y)
1470 IF (X=0)+(Y=-4)=-2 THEN
 1510
1480 NEXT I
1490 CALL POKEV(799,C1)
1500 GOTO 1400
```

LISTINGS

```
1510 CALL POKEV(799,C1)
1520 Z1=5
1530 S1=24
1540 Z2=8
1550 S2=24
1560 GV=142
1570 GOSUB 3090
1580 FOR I=1 TO 2
1590 CALL KEY(1,K,ST)
1600 IF K=18 THEN 1820
1610 CALL JOYST(1,X,Y)
1620 IF (X=0)+(Y=0)>-2 THEN
1730
1630 NEXT I
1640 CALL POKEV(799,C2)
1650 FOR I=1 TO 2
1660 CALL KEY(1,K,ST)
1670 IF K=18 THEN 1820
1680 CALL JOYST(1,X,Y)
1690 IF (X=0)+(Y=0)>-2 THEN
1730
1700 NEXT I
1710 CALL POKEV(799,C1)
1720 GOTO 1580
1730 CALL POKEV(799,C1)
1740 IF SP=2 THEN 1760
1750 IF ABS(X*Y)=16 THEN 158
0
1760 Z2=Z1-(SGN(Y)*(ABS(Y)-1
))
1770 S2=S1+(SGN(X)*(ABS(X)-1
))
1780 IF (Z2<8)+(Z2>23)THEN 1
580
1790 IF (S2<3)+(S2>30)THEN 1
580
1800 GOSUB 3090
1810 GOTO 1580
1820 CALL POKEV(799,C1)
1830 X=(S1-3)/3+4
1840 Y=(Z1-8)/3
1850 BO=BO-FA
1860 IF BO<=0 THEN 2560
1870 CALL LINK("D",162,P$(BO
))
1880 IF S(X,Y)>0 THEN 1930
1890 ON S(X,Y)+5 GOTO 2040,2
120,2280,2420,1900
1900 GOSUB 3150
1910 CALL SOUND(-150,220,0)
1920 GOTO 2020
1930 GOSUB 3150
1940 FOR I=1 TO S(X,Y)
1950 CALL SOUND(-200,1000,0,
-5,9)
1960 CALL LINK("O",I)
1970 CALL SOUND(-800,-5,9)
1980 GOSUB 2960
1990 NEXT I
2000 SC=SC+150*S(X,Y)
2010 CALL LINK("D",66,P$(SC)
)
2020 CALL POKEV((Z1-1)*32+S1
-1,224+S(X,Y),"",151,249)
2030 GOTO 1400
2040 GOSUB 3150
2050 FOR I=1 TO 6
2060 CALL LINK("O",I)
2070 CALL SOUND(-10,220,0)
2080 NEXT I
2090 CALL LINK("O",0)
2100 CALL POKEV((Z1-1)*32+S1
-1,221,"",151,249)
2110 GOTO 1400
2120 GOSUB 3150
2130 FOR I=1 TO 3
2140 CALL SOUND(150,262,0)
2150 CALL LINK("O",I-1)
2160 CALL SOUND(150,440,0)
2170 CALL LINK("O",I)
2180 CALL SOUND(150,262,0)
2190 CALL LINK("O",I+1)
2200 CALL SOUND(450,349,0)
2210 CALL LINK("O",0)
2220 NEXT I
2230 BO=BO-SP*500
2240 IF BO<=0 THEN 2560
2250 CALL LINK("D",162,P$(BO
))
2260 CALL POKEV((Z1-1)*32+S1
-1,222,"",151,249)
2270 GOTO 1400
2280 GOSUB 3150
2290 CALL HCHAR(Z1,S1,135)
2300 FOR I=1 TO 3
2310 FOR J=110 TO 180 STEP 5
2320 CALL SOUND(-100,J,0,J+1
,5,J+2,10)
2330 NEXT J
2340 NEXT I
2350 CALL LINK("D",138," FAL
LGRUBE! ")
2360 CALL LINK("D",162,"0000
0")
2370 GOSUB 3360
2380 CALL LINK("D",138,"ssss
ssssssss")
2390 GOSUB 2960
2400 BO=0
2410 GOTO 2560
2420 GOSUB 3150
2430 FOR I=1 TO 5
2440 CALL LINK("O",0)
2450 FOR J=1 TO 50
2460 NEXT J
2470 CALL SOUND(100,440,0,69
8,0)
2480 CALL LINK("G")
2490 FOR J=1 TO 50
2500 NEXT J
2510 NEXT I
2520 SC=SC+5*FA+500*SP
2530 CALL LINK("D",66,P$(SC)
)
```

```
2540 CALL POKEV((Z1-1)*32+S1
-1,239,"",151,249,"",59,144+
GOLD)
2550 NEXT GOLD
2560 SC=SC+BO
2570 IF SC<99999 THEN 2590
2580 SC=99999
2590 IF HS>SC THEN 2640
2600 HS=SC
2610 CALL LINK("D",138," HIG
HSCORE! ")
2620 GOSUB 3380
2630 H$=SEG$("00000",1,5-LEN
(STR$(HS)))&STR$(HS)
2640 CALL LINK("D",66,P$(SC)
)
2650 CALL LINK("D",130,"HIGH
S")
2660 CALL LINK("D",162,H$)
2670 CALL POKEV(785,16)
2680 CALL LINK("C")
2690 FOR I=1 TO 3
2700 GOSUB 3340
2710 NEXT I
2720 X=74
2730 Y=110
2740 GOSUB 2990
2750 IF (K=78)+(K=110)<0 THE
N 2890
2760 IF (K<>74)+(K<>106)=-2
THEN 2720
2770 CALL CLEAR
2780 SC=0
2790 FOR Y=0 TO 5
2800 FOR X=0 TO 17
2810 W(X,Y)=0
2820 NEXT X
2830 NEXT Y
2840 FOR I=0 TO 14
2850 D1(I)=0
2860 D2(I)=0
2870 NEXT I
2880 GOTO 250
2890 FOR I=1 TO 5
2900 CALL LOAD(32047+I,ASC(S
EG$(H$,I,1)))
2910 NEXT I
2920 CALL LINK("E")
2930 X=INT(RND*10)+4
2940 Y=INT(RND*6)
2950 RETURN
2960 FOR L=1 TO 200
2970 NEXT L
2980 RETURN
2990 CALL KEY(0,K,ST)
3000 IF (K>=X)+(K<=Y)=-2 THE
N 3080
3010 FOR L=1 TO 15
3020 CALL SOUND(-300,-5,15)
3030 CALL KEY(0,K,ST)
3040 IF (K>=X)+(K<=Y)=-2 THE
N 3080
```

LISTINGS

```
3050 NEXT L
3060 CALL SOUND(-1000,1000,5
,-5,15)
3070 GOTO 2990
3080 RETURN
3090 GH=GV
3100 CALL GCHAR(Z2,S2,GV)
3110 CALL POKEV((Z1-1)*32+S1
-1,96+GH,"",(Z2-1)*32+S2-1,2
49)
3120 Z1=Z2
3130 S1=S2
3140 RETURN
3150 CALL LINK("O",0)
3160 CALL POKEV(136,238,"",1
68,238)
3170 CALL SOUND(-4000,110,16
,-5,9)
3180 FOR L=1 TO 6
3190 T=INT(RND*7)
3200 CALL LINK("O",T)
3210 CALL POKEV(785,145)
3220 FOR V=1 TO 50
3230 NEXT V
3240 CALL LINK("O",0)
3250 CALL POKEV(785,17)
3260 FOR V=1 TO 50
3270 NEXT V
3280 NEXT L
3290 CALL POKEV(136,237,"",1
68,237)
3300 RETURN
3310 T=INT(RND*7)
3320 CALL LINK("O",T)
3330 RETURN
3340 RESTORE 3450
3350 GOTO 3390
3360 RESTORE 3460
3370 GOTO 3390
3380 RESTORE 3520
3390 READ L
3400 FOR J=1 TO L
3410 READ L,TN,V
3420 CALL SOUND(L,TN,V,TN+2,
V+5,TN-2,V+5)
3430 NEXT J
3440 RETURN
3450 DATA 2,150,800,5,150,60
0,5
3460 DATA 23,500,131,0,500,1
31,0,1000,175,0,50,30000,25
3470 DATA 500,131,0,500,175,
0,800,220,0,200,196,0
3480 DATA 200,175,0,200,196,
0,500,220,0,200,196,0
3490 DATA 200,175,0,200,196,
0,800,220,0,200,30000,25
3500 DATA 800,175,0,500,220,
0,350,175,0,800,131,0
3510 DATA 500,131,0,500,131,
0,1200,175,0
```

```
3520 DATA 8,500,220,0,500,33
0,0,1000,440,0,250,220,0
3530 DATA 250,330,0,250,220,
0,250,440,0,1000,550,0
```

Fortsetzung von Seite 46

Die Bildschirmanzeige informiert über den Score, den Bonus, die Zahl der gefundenen und zu suchenden Goldstücke, ob mit oder ohne Detektorstörfeld und Fallgrube gespielt wird.
Der Score für jeden Zug und für gefundene Goldstücke richtet sich nach dem Schwierigkeitsgrad des Spieles. Der Bonus nimmt für jeden benötigten Zug entsprechend dem Schwierigkeitsgrad ab.
Der höchste Score ist nur mit dem größten Schwierigkeitsgrad erreichbar: komplexes Spielfeld, 6 Goldstücke, mit Detektorstörfeld und Fallgrube!
Zur Eingabe: Die Punkte in Zeile 160 von Goldsucher II sind wie üblich als Leerzeichen einzugeben, ebenfalls diejenigen in den Data-Zeilen von Teil 1, bzw. immer, wenn zwei oder mehr Punkte direkt hintereinander stehen!!!

Reiner Makohl

BÖRSE

Mich würde einmal interessieren, ob es im Raum Leonberg und Umgebung noch andere TI-User gibt. Es würde mich freuen, wenn sich jemand meldet. Tel: 07156/34941 (Carlos).

Verk. Espial 35,– DM, Car Wars 20,– DM, Tomb City 20,–, Attack 20,– DM, Soccer 25,– DM. Suche Star Treck. M. Kreher, Tel: 06055/4684.

Verk. original TI-Module: Pa Parsec, Car Wars, Munchman, Attack, Tombstonecity, je 40,– DM. Orig. Dynamicspr. Digger, Monoko, Darts. Holger Wesemann, Kreuzkrug 62, 3079 Raddestorf 1.

Verk. geg. Höchstgebot: TI 99/4 A m. Ex.-Basic, Joyst., sämtl. Zubehörkabel, sehr viel Literatur u. Module. Tel: 0203/364897, Schulz.

Schüler sucht billig: Ex.-Basic Adventure-Modul und Software. Harry Sander, Fri.-Hebbel-Str. 15, 2222 Marne.

Verk. TI 99/ 4 A, Ext.-Basic, TI-Rec., Joyst., 350,– DM. TI-RS232 (V24) Schnittstelle Anschl. Kabel Epson-Drukker 180,– DM, viele Arbeitsmodule je 30,– DM, Spielmod. je 20,– DM. M. Rauhut, Meisenweg 19, 4173 Kerken.

HELP!!!HELP!!!HELP!!!
Ich habe das Adventure Return to Pirate's Isle und komme einfach nicht weiter. Wer kann mir Tips geben??? Tel: 07156/34941. C. Jarque.

VERSCHENKEN kann ich leider nichts, jedoch verk. ich folgende Module billig: Soccer, Tomb, City, Yahtzee, Munchman, Car Wars, Alpiner, Miner2049er, The Attack, A-maze-ing, Buchungsjournal, Statistik Datenverw., Connect four Speech Editor. Sprach Synthesizer, Staubschutzhaube, orig. Diskettensoftware von FUTURA, DSK-Manager, E/a-Modul.
Alle Preise VB. Wer zuerst kommt, mahlt zuerst!!!
Tel: 07156/34941 (Carlos).

Last Mission, das größte Adventure für den TI + Adventuremodul! Es besteht aus: 127 Meldungen, 105 Objekten, 112 Substantiven, 55 Orten und 69 Verben!
30,– DM beilegen, Datenträger angeben und ab an: CCK, Am Hellenberg 24, 6367 Karben 1.

Werdet Mitglied im Computerclub Karben. Beitrag für 3 Monate 4,– DM! Fordert Info (–,80 DM) oder Clubzeitung (2,– DM) an: CCK, Am Hellenberg 24, 6367 Karben 1.

Verk. 32 K-Erw. est. 200,– DM und MBI Interface 150,– DM. Tel: 0221/173274.

Verk. TI 99/4 A kpl. + X-Basic + Handb. + Doppelcass. Kabel + 15 Cass. + massig Literatur + Software satt!!! (Hardware orig. TI). VB 400,– DM. Mehr Info bei: J. Müller, Espanstr. 84, 8510 Fürth. Tel: 0911/799425.

Verk. TI 99/4 A 250,– DM, Diskl. 225,– DM, 32 KB 225,– DM, Module, Bücher, Spiele u. Listings aus alten Zeitschriften. Angebote und Anfrag. an: Markus Theimer, Schwedenstr. 2, 6203 Hochheim.
Rückumschlag bitte beileg.

Suche Schaltplan für Discontroller sowie DSR-ROM-Progr. und Diskmanager. Biete Technical Data Handbook für Konsole und Ex-Box oder Bares. M. Wahner, St.-Georg-Str. 37, 3100 Celle, Tel: 05141/25990 nach 19 Uhr.

Verk. Minimem 130,– DM, Pers. Rec. Keeping 60,– DM, Statistik 60,– DM, Adventure mit 1 Spiel 50,– DM, Grafiktablett 100,– DM. Tel: 030/6262950 oder Scheck an: L. Kaschke, Wederstr. 59, 1000 Berlin 47 - Lieferung erfolgt sofort.

Verk. TI 99/4 A m. div. Hardware u. Modulen. Preisliste anfordern bei: Jörg Geisser, Obervorschützenstr. 34, 3505 Gudensberg.

Verk. ID-30 25,– DM, ID-Data 25,– DM, ID-Pilot 25,– DM, Oldies but Goodies, je 10,– DM, Basic für Anfänger 10,– DM, Minimem. 160,– DM, Dateiverw. Modul 50,– DM, Munchman 25,– DM, Parsec 30,– DM. Michael Strub, Hauptstr. , CH-8774 Leuggelbach.

ENDLICH! Ähnl. Suchroutine für Basic-Dat.-Verw.Progr. Schluß mit d. Tippfehlern u. unklaren Schreibweisen. Diese Routine findet alles! Info geg. frank. u. adr. Rückums. bei: T. Beuck, Horner Redder 14, 2000 Hamburg 74.

Verk. 2 TI 99/4 A-Konsolen 1 Ex.-Basic + Handbuch deutsch, orig. TI-Joystick, Invaders + Pool Pos., 10 Cass. voll m. Progr. Alle TI-Revues. NP 1400,– DM, kpl. 700,– DM. Tel: 02241/42978.

CCK, Am Hellenberg 24, 6367 Karben 1. Wir bieten: Clubzeitung, Programmtausch, Programmierhilfen, Berichte. Sonderpreis für Mitglieder von Firmen. Fordert: Info –,80 DM, Clubzeitung 2,– DM.

Verk. Assemblererweiterung von Radix. VB 100,– DM. Jochen Schmitt, Am Hellerberg 24, 6367 Karben 1, Tel: 06039/2772.

Verk. TI 99/4 A, Ext.-Basic, Munchman, Bücher, TI-Revue, Rec.Kab., zus. VB 400,– DM. Auch einzeln! Holger Scherer, Jungstr. 25, 6550 Bad Kreuznach.

Wer tauscht Module geg. Cass. Software? Angebote an: Manfred Schneider, Nunkircher Str. 42 a, 6648 Wadern-Büschfeld/Saar, Tel: 06874/6705.

Schnelle Kassettenroutine! Speedtape 2000 (Baud). Progr. f. Kons., 32 K + MM. MM. Operiert im ganzen CPU-Ram. Bis 32 K-Progr. lad- u. speicherbar! + Basicroutine. Progr.-Kass. + Anl. 50,– DM Scheck an: Rudolf Mauk, Justgasse 29/67/1, A-1210 Wien, Tel: 0222/3974462.

Kontakte
Tausche Apesoftprogr. (XB-2plus), natürlich auch XB u. TIB. Michael Bolte, Am Sportplatz, 3065 Nienstädt, Tel: 05721/2627.

Wer kann mir weiterhelfen? An alle TI-User-Clubs u. Software-Freaks! Suche für TI 99/4 A Cobol-Compiler. Wer weiss näheres darüber? Gibt es den Compiler überhaupt? Bitte meldet Euch!!!
D. Voges, Schönhauserstr. 11, 4100 Duisburg 1, Tel: 0283/370104.

Verk. 32 K CMOS, RS232, Laufw. intern (neu). Kaufe Anl. für 3D-World, ID Data/ Konto, Ed./Ass., TEZ, Multiplan. Suche Adventures auf Disk. Tel: 07243/78582.

TI 99/4 A Super-Angebote: Konsole 150,– DM, Editor-Ass.-Paket 120,– DM, Disk-Kopierer Super-Duper-Modul 80,– DM, Speech-Synth. 120,– DM, Minimem. 160,– DM, Alpiner und Othello je 25,– DM, Parsec 45,– DM. Tel: 08141/23140 nur 18 bis 21 Uhr.

BÖRSE

Suche einen in Assembler geschriebenen ASSEMBLER für das X-Basic (auch Pokeliste in Basic) auf Cass. Tausche PRGRM'S. Tel: A/0662/26671.

Verk. Buch TI-Intern 25,– DM, TI-Special 25,– DM, P.-Box + 32 K + RS232 + Disk Cont. + 2 Laufw. 2300,– DM. Tel: 02152/50188.

Suche Sprachsynthesizer (mit Anleitung) und gut erhaltene Bücher: Editor Ass. Handbuch, deutsch, 99 Special 1 + 11, TI 99/4 A Intern. Jens Otten, Senefelder Str. 15, 6200 Wiesbaden.

Verk. TI-Writer, Multi Plan, Mini-Mem, TE-II, ED/Ass, Text u. Datei, Datenv./Ana., + div. Spielmodule. Tel: 064 41/74830.

Centronics-Extern – neu – mit deutsch. ausf. Anleitung. Ohne Kabel, original TI-Kabel verwendbar. VB 120,– DM. Dierk Warburg, Lilienweg 12, 7141 Benningen, Tel: 071 44/7558.

Verk. 8 verschied. Slott-A. Adventure (Originale), teilweise doppelt, auch Tausch! Tel: 07144/7558.

Tausche Modul Miner 2049er gegen Modul Shamus evtl. Wertausgleich. Thomas Külpmann, Lützowstr. 54a, 5800 Hagen, Tel: 02331/21454

Suche TI-Write-Disk mit deutsch. Handbuch. Verk. 2 engl. TI-Ex1-Handbücher und Superspiele für den TI. Tel: 0711/843775

Verkaufe: Mini Assembler/ (Radix) kompl. mit Software für 95,– DM. Bei: H.W. Hansen, Tel: 06897/76 4325

Achtung! Wer hat Lust mit mir Programme zu tauschen? Ich besitze EX, 32 K, DSK, ED/As. Also schickt Eure Tauschlisten an: Marco Wintzer, Mühlstr. 42, 6070 Langen.

Verk. Orig. Ext. Basic, Spiele in Ext. auf Cass., deutsch. u. engl. Handb. Kompl. VB 170,– DM, Tel: 06441/52 556

Suche def. Comp. u. Zubeh. sowie Schaltp. auch Farb-TV Vidio usw., W. Jendrock, Lippestr. 24, 4350 Recklinghausen

---

Suche Terminal Emulator (TE II). Verkaufe Speech Editor (30,– DM. Suche Kontakt zu TI-Fans, Raum Stgt. Tel: 07026/5885.

Verk. TI-Progr. für Bauwesen. Info. 069/818451.

Verkaufe: Spielmodule Dig-Dug Defender, TI-Invaders, Tomostone-City-Wumpus. Preis VB, Tel: 06047/4779.

Achtung Adventure-Freaks! 6 neue Adventures (Eigenentwicklung) für Adventure Modul. Info gegen frankierten Rückumschlag bei: JIM-SOFT, z. Hd. Hörg Illerich, Euskirchner Str. 23, 5000 Köln 50.

Verkaufe Mark Plan. Old B. Gol, Finanzb., Basiclehrg., Minus Mission, je 20,– DM. Tel: 06134/4519

Verkaufe TI 99/4 A 3 KEC. Kabel + Recorder + Schachmodul. Preis: 350,– DM. Angebote an: E. Driefert, Düsseldorfer Str. 114, 4050 Mönchengladbach 2.

TI-99 Club der Toten VC 20 Wir suchen noch Mitglieder! Der Club der Toten (CDT) bietet: Softwarebibliothek, Einkaufsgemeinschaften, Clubinfo, nach Wunsch Clubtreffen und seit neuestem: 2099er-DAS CIMPUTERMAGAZIN DES CLUBS DER TOTEN (nicht nur für Mitglieder). Clubinfo gegen Rückporto. Adresse: Club der Toten, Am Hang 5, 2054 Geesthacht.

Tippe Listings für Sie ab (aber nur TI-Basic) 9,– DM + Kassette von: Stefan Schwarz, Dürerstr. 22, 4630 Bochum 1.

Verkaufe kopierte Listings. Info bei: Erwin Kinslechner, Eibesbrunnergasse I/10/4, A-1120 Wien.

We are looking fpr TI-Fans all over the world!!! Erwin Kinslechner, Eibesbrunnergasse 1/10, A-1120 Wien.

Suche Drucker!!! Erwin Kinslechner, Eibesbrunnergasse 1/10/4, A-1120 Wien.

Wer hat Minigolf??? Erwin Kinslechner, Eibesbrunnergasse 1/10/4, A-1120 Wien.

Verk. 32 K-Karte. Tel: 0809/ 12777.

!!!SUPER-ANGEBOT!!!
92 Basic-Spiele für nur 70,– DM! Tel: 08142/52873.

TI 99/4 A. Kompl.-System und Profi-Software mit ausführlicher Literatur. A-Tel: 0316/705844

Verk. TI 99/4 A + X-Basic +Minimem + Recorder + Kabel + Joystick + 5 Handbücher für 500,– DM. C. Bobek, Tel: 06752/2752 ab 18 Uhr.

Suche TE 1200 und Multiplan auf Disk. H. Vetter, Heidekampstr. 9, 4800 Bielefeld.

Verk. Module: The Attack 15,– DM, Speech Editor 20,– DM, Pers. Rec. Keep 20,– DM, Music Maker 40,– DM. Tel: 04841/4435.

Verk. TI 99/4 A + Joystick + Disk. Laufwerk + Ex.Basic + Disk.Manager + Controller ext. + div. Spiele + Programme auf Disk. + Deutsche Handbücher, VB 1500,– DM. Tel: 05255/6783, Reimann.

Verk. TI-P-Box mit Diskkont. u. Laufwerk, V 24, 64 K Bausatz, zusammen VB 950,– DM. Tel: 05323/1811.

Für TI 99/4 A, Buchungsjournal 110,– DM, Disk Canon MDO 211 (BASF 6128) 40SP-DS 300,– DM, Disk BASF 6106 40/ss 250,– DM, neuw. Tel: 06405/3511, Freitag.

TI 99/4 A + Rec. Kabel + Buch TI-Intern + Modul Exb., Buchhaltg. + Cass. TI-Basic, ExBasic + alle TI-Revue + weiterer Literatur, 300,– DM Tel: 089/938973, vormittags

---

MBI-Centronic + Rec.Kabel + Buch TI-Intern + Buchungsjournal, alle TI-Revue, 250,– DM. Tel: 089/938973 vormittags.

Verk. Sharp-Poket-Comp. 1251 mit Druck. + Rec. + Accu-Opt. CE 125, Netzger. LiT., Inter-F.F. Ext. Rec., kompl. 400,– DM, Tel: 0841/74205, nach 19 Uhr, Otzko.

Suche dringenst Ex.B. Günter Simone, Baslergasse 50-66/43/1, 1232 Wien.

Suche TI-Freunde im Raum Wien!!! Günter Simone, Baslergasse 50-66/43/1, 1232 Wien, A.

Verk. Schachmod. 80,– DM, Attack 30,– DM, Statist. 60,– DM
RS232ext. 240,– DM, def. Kons. VB. Tel: 0413/155457

Verk. TI 99/4 A + orig. Joyst. + Ex-Basic + Module, The Attack, Blastot Car Was + Munchman + Othello + Parsec + Datenverw. u. Analysedatei + Textprogramm + Sprachsynth. Tel: 0209/ 56800, P. Ruth.

Biete/Suche/Tausche: Hardw. einzeln/kompl. + orig. Software. Tel: 0572/6409 ab 17 Uhr, G. Bürger, Auhagen.

Verk. X-Basic-Progr. ab 0,50 bis 5,– DM. Info geg. Freiumschlag. Ralf Bauer, Kelterstr. 16, 6962 Ad.-Sennfeld.

Verk. P-Box für den TI 99/4 A, VB 300,– DM. Tel: 05181/6487 nach 14 Uhr.

Schweiz: Minimem 120,– Fr Centronics/Bus/Kabel 120,– Fr, Statistik 30,– Fr, Parsec/Chis/Alpiner je 20,– Fr. 065/352816.

Verk. TI 99/4A kompl. + X-Basic + Handbuch + Doppelcass. Kabel + 15 Cass. + 8 TI-Revue + TI 99 intern + Sonderhefte + Software satt! VB 400,– DM! Tel: 0911/799425, Müller.

!!!Hilfe!!!
Ich suche Extended-Modul und Handbuch, sowie Programme für Lottsysteme. Erich Glöckle, Pf. 1922, 7900 Ulm.

Alle Zahlensysteme TI + Ext. Basic Kass., 15,– DM. Ralf Feldhoff, Hüttenstr. 37, 4000 Düsseldorf 1, Tel: 0211/377795.

Suche: F. TI 99/4 A P.Box mit RS 232 32 K Disc. Contr.Laufwerk Disc Managermodul. Tel: Frankreich 88/962666, Wendling.

Super Gelegenheit: TI 99/ 4 A mit Ext.Basic, TI-Invaders, TI-Haushaltsmodul a. Orig. + Bücher + Spiele. Tel: 08151/12089, Schmid.

Wir suchen TI-Fans in der ganzen Welt!!! Erwin Kinslechner, Eibesbrunnergasse 1/10, A-1120 Wien.

Wir suchen TI-User zwecks Programmaustausch. Erwin Kinslechner, Eibesvrunnergasse 1/10/4, A-1120 Wien.

BÖRSE

Verk. 32 K Byte-Erweiterung für nur 170,– DM (CMOS-Technik, mit Batteriepufferung, seitl. ansteckbar). Jens Otten, Senefelder Str. 15, 6200 Wiesbaden, Tel: 06122/700239, Samstag von 18 bis 20 Uhr.

TI 99/4 A + 32 K + Ext.Basic + Int. Floppy + Literatur, VB 1400,– DM. Verk. auch einzeln. Tel: 0214/401384 von 17 bis 20 Uhr.

Suche Moonsweeper od. Espial o. Microsurgeon! Zahle 30,– DM! Parsec o. Su. De. Attack, 20,– DM. Angebote: D. Augustin, von Hasewinkelweg 87, 5020 Frechen 4.

Verk. Ext-Basic + deutschem Handbuch. Suche Rechenk.-Mod. Suche Mini-Mem.-Mod. Tel: 04554/6396.

Verkaufe/Tausche (geg. RS232-Karte): Centronics ext. + 4 Module. Tel: 06821/7522, Oliver Siffrin.

Nappsoft tauscht Programme in TI u. Ex.-Basic. Liste an: M. Roth, Brucknerstr. 3, 6680 Neunkirchen 7.

Verk. Module: Ex.-Basic 120,– DM, Mini-Mem 170,– DM + Literatur. Disk, Fixer 80,– DM, TE 2 50,– DM, Tel: 069/783480.

Suche Sprachsynthesizer für max. 66,– DM. Tel: 07134/15276, A. Hein.

Verk. Schachmodul 50,– DM und Return to Pirates Isle 70,– DM. Michael Buresch, Am Friedhof 23, 2200 Elmshorn.

Verk. Zeitlupenmodul (I/0-Port) für TI 99 mit Geschwindigkeitsregelung und Standbild für nur 30,– DM. Michael Buresch, Am Friedhof 23, 2200 Elmshorn, Tel: 04121/85363.

Verk. Ext. 32 K -Erw. (Batteriegepuffert) für 140,– DM. Michael Buresch, Am Friedhof 23, 2200 Elmshorn.

Tausche Programme in Exrended Basic II plus 1seitig bespielte Cass. an: Michael Bolte, Am Sportplatz, 3065 Nienstädt.

Verk. TI 99/4 A Ex,-Basic + 32 K, Rec. + Kabel 480,– DM, Alpina invadas, Pars., E. Reading, Bigfoot Dat. D.MA. usw., Tel: 040/6950133.

Verk. orig. TI-Joysticks u. Datenrecorder für je 30,– DM. Michael Buresch, Am Friedhof 23, 2200 Elmshorn, Tel: 04121/85363.

Lösung zu Pirate's Isle. Nur 5,– DM (Schein, Scheck). Sven Porstendorfer, Brahmsstr. 29, 2200 Elmshorn.

TI 99/4 A mit Recorderkabel f. zwei Rec., Speech-Synthysizer, orig. Joysticks, Recorder, Module: Ext.-Basic, Statistics German, Schachmeister, Household Budget Management, Parsec, Tunnels of Doom, viele Kass., Bücher Zeitschr., Preis: 870,– DM. Bernd Stumpf, Kai Friedrich Str. 80/81, 1000 Berlin 10, Tel: 030/3418884.

!!!Assembler!!!

Verk. Miniassembler der Fa. Radix. VB 110,– DM. Lukas Merten, Marktstr. 18, 5440 Mayen.

Suche Amateurfunkprogramme für MSX LX1CW. Gilbert Schockweiler, Pf. 185, 9002 Ettelbruck, Luxembourg.

Suche Floppy + Drucker für MSX. Biete günstig Software. Markus Giel, Breslauerstr. 33, 4353 Oer-Erkenschwick.

TI 99/4 A, Ex-Basic, Schach, weitere Module, jede Menge Software u. Literatur, Speech usw. Preis VB. Tel: 030/3418884.

Tausche Microsurgeon-Modul gegen TI-Invaders-Modul. Tausche Schachmodul gegen Adventure-Modul + mind. 2 Adventure's. Tel: A/0662/26671.

Disketten-Editor für E/A, XB oder MM + 32 K. Viele Funktionen, menügesteuert. Dazu 4 Seiten Info über Datenspeicherung auf Disk. Damit knacken Sie jedes Problem! Info gegen Freiumschlag. Mathias Eichhorn, Ziegelheck 1, 6240 Königstein 4 (Schneidhain), Tel: 06174/5921.

TI-Module: Invaders, Carwars, Chisholm-Trail, Attack, je 40,– DM. Tel: 0421/256378.

Extended Basic II plus Suche Kontakte im In- und Ausland zwecks Programm- und Erfahrungsaustausch. Listen oder 1seitig bespielte Cass. an: Michael Bolte, Am Sportplatz, 3065 Nienstädt 1

Achtung! Suche TI-User im Raum Lohr und billiges Ex-B. für max 100,– DM! Thomas Pfister, Talstr. 49, 8770 Lohr/Main.

Verk. Ex.-Basic + Handbuch 120,– DM. Tel: 05252/51869.

Invaders 30,– DM, Tombst. 30,– DM, Video 1 20,– DM, Attack 30,– DM, Parsec 40,– DM, Chess 40,– DM, Soccer 20,– DM, Musicm. 30,– DM. Tel: 05546/1088.

Programmierbarer Taschenrechner TI 66 günstig zu verkaufen, für nur 100,– DM. Tel: 05546/1088.

Advent. Mod. u. 6 Cass. 110,– DM, Dig Dug 36,– DM, Fitness 30,– DM, Depender 35,– DM, Alpiner 25,– DM. F. Lederer, Wendelinusstr. 39, 7520 Bruchsal 4:

Verk. Mini-Mem. + Anleitung + Assemblerbuch, 180,– DM. TI-Modul invaders 20,– DM, zusammen 190,– DM. Tel: 069/783725.

Verk. TI 99/4 A + Ext.-Basic + Parsec + Schach + Yahtzee + Moonsweeper + Statistik + Kabel + Software + TI-Revue, kompl. für 499,– DM, ab 3.5. Tel: 08092/2915.

Verk. neues Parsec-Modul ohne Anl. für 20,– DM. Michael Schneller, Tel: 08233/5412.

Suche Tauschpartner in TI-Basic. Tel: 02592/61816.

Verk. od. tausche Miniassembler v. Radix, neu!!! Musik-Maker + Finanzhilfe, beide Originale, zusammen 110,– DM. Tel: 08841/40346.

Suche: TI-Video-Chess, bis 50,– DM und TI-32 K-Erw. (extern) auch m. Centronicschnittstelle. Tel: 0201/674076 ab 15 Uhr.

Verk. TI 99/4 A + Ext. Basic + orig. TI Joyst. + Cas.Rec. Kabel + Basic-Lerncass. + Datenverw. + Statistik + 5 Spielmodule + 2 Bücher + alle TI-Revue, nur 550,– DM. Tel: 02687/539. Marco Kley, Hauptstr. 19, 5451 Pleckhausen.

Suche Drucker. Biete Programme nach Wunsch + Schachmodul + Money. Tel: 0211/377795

Tausche Microsurgeon + Moonsweeper geg. gut erhaltenes TE II + Handbuch. Tel: 06134/4519, Dirk Junghans.

Das Kochbuch für den TI, Band 1 10,– DM. Cass.Rec. Kabel 10,– DM. Tel: 06638/1503.

Verk. TI 99/4A + Ex.-Basic + 32 K + Centr.m.Kabel + S-Synth. + Rec.m.Kabel + Joystick + Module: Statistik Datenver. Add. Min. Othello, + TI-Intern + Progr-Cass. TI/1-8 + Literatur + Progr., zus. 500,– DM. Reinhard Püttmann, Fichtenstr. 26, 4100 Duisburg 29.

Verk. für TI 99 Siemens Floppy mit Diskkontroller. Ruft S. Altstadt an: Tel: 07253/22681.

Suche für TI 99/4 A 32 K-Erw. evtl. Tausch gegen andere Periph. Ruft S. Altstadt an: Tel: 07253/22681

Suche Cartridge-Expander bis ca. 80,–DM. Angebote schriftlich an: Walter Beez, Reinhardstr. 40, 6000 Frankfurt/M. 50.

TI Modulator, CH 4, 40,– DM, Netzteil Trafo 25,– DM Defect Hauptplatine 40,– DM Ed./Ass.-Buch 40,– DM, TI 99/4A Intern 25,– DM. Tel: 06155/4692.

Verk. Module: Invaders 15,– DM, Football (USA Modul) 25,– DM, Tunnels of Doom 60,– DM, Mini-Mem. 100,– DM, Adventure + 11 Adventures 80,– DM, Cass. Kabel 10,– DM, TI Tips + Tricks 10,– DM. Tel: 06421/44408 bei Vale.

TI 99/4 A + Palmodulator + Adapter für 220,– DM. Der Preis ist VB, allerdings inkl. Tibasic, Extendedbasic-Programmierhandbuch + Recorderkabel. Tel: 0203/583718.

Raum Karlsruhe! Suche Kontakt zu Besitzern des Extended Basic II Plus zwecks Erfahrungsaustausch. Tel: 0721/885216, W. Merkelbach

---

Verk. P.-Box, Laufw., Contr., Manager 1350,– DM, RS232 250,– DM, 32 K 250,– DM, High screen Monitor 250,– DM, Ext.Bas. 150,– DM, Soccer 30,– DM, Carwars 20,– DM, Othello 10,– DM' TI 99/4 A 100,– DM. 05121/62716.

BÖRSE

Alle die im Raum Ludwigsburg/Stuttgart einen TI haben, gönnen diesem jetzt eine Pause. Melden: EA/XB/DK. Dierk Warburg, Lilienweg 12, 7141 Benningen. Tel: 07144/7558.

MBI multiboard Interface TI 99/4 A auf Centronics anschlußfertig mit Kabel und Stecker, aber Bus nicht durchgeschleift für 120,– DM zu verk. B. Westermann, Gutenberstr. 20, 6710 Frankenthal, Tel: 06233/20306.

Suche preisg. deutsch. E/A-Handbuch u. Drucker mit CentroMics-Interface. Tel: 02173/57366.

Neuwertig! Original TI-Ex-Basic Modul (LTA 1184 USA) mit original Handbuch sowie deutschem Handbuch, 190,– DM, Tel: 06249/4419

Verk. Kompl. anschlußfertig TI 99/4 A + Ex-Basic + Cass.-Recorder m. Kabel + Dateiverw. + Soccer + orig. TI-Joysticks + Bücher + Progr. + TI-Journal/Revue, VB 500,– DM. Tel: 0431/725 489.

TI 99/4 A

zur verkaufen- Systemwechsel Konsole kpl. + Ers. Tastatur Orig. TI Ex-Basic + Handbuch COS99 Peri.Box inkl. DD, Disk (Epson) a360kb, Controll. 1xV24 + 2x RS232 Schnittst. + 32kb Erweiterung+V24 Kabel, Datenrck. incl. 2x Kabel, RGB-Modulator, Sprach-Synth, Orig. TI-Joyst., 2 x Joyst. Org Quickshot II inkl. Adapter Disk Progr: Orig. TI Rechn. Stellung, Lagerverw., Versandl., Module: Datenverw., Statistik + Buchungsj. dtsch. sowie Video Games I + Othello + Chish. Trail, Cass.-Progr.: TI-Basic + TI-Ex-Basic + Oldies but Goldies I + II Finanzberater + Marketing Planspiel + Dynamics Cass. Monoko, Darts, Digger, Zentoria, Schallpl.-Datei, Lager-Datei, Vokabel Train. Jede Menge Fachliteratur und TI-Journale einschl. TI-Journal Sonderbände: 1 + 2 sowie Assembl. Heft. NP über 6000,– DM, Kpl. Abnahme VB, CPS System noch Garantie. Tel: 02234/74448 ab 19 Uhr.

Schnellste Datenverw. mit Kass. Ex-Basic-Datenverw.-Progr. auf Kass. geg. Einsend. von 10,– DM. Peter Hielscher Am Wall 22, 4401 Saerbeck.

R.G. Softgame, Spitzensoftware in TI o. Ex-Basic. Info für 1,50 DM bei R. Goerke, Watzling 10, 8250 Dorfen.

Suche externe Disk + Cont. Tel: 08081/3236

TI-Bücher: TI-Intern (Martin) 20,– DM, TI-Spezial I 30,– DM, TI-Spezial II 40,– DM, alle Bücher neuwertig. Peter Hielscher, AmWall 22, 4401 Saerbeck.

TI-Module jeweils 30,– DM. Hustle, Amazeing, Hangman, Videospiele II, Connect Four, Kniffel, Othello.
TI-Module jeweils 50,– DM. Statistik, Datenverw. + Analyse, Adventure. Peter Hielscher. Am Wall 22, 4401 Saerbeck.

TI-Kassetten: Grundkurs 10,– DM, Finanzber. 20,– DM, Marketing-Plansp. 20,– DM, Computer-Kontakt TI 1A-Kass. 20,– DM. Modul: Music-Maker 50,– DM. Peter Hielscher. Am Wall 22, 4401 Saerbeck.

Verk. Mini-Mem. + Demo-Cass. orig. TI.+ Literatur 190,– DM o. tausche geg. Sprachsythesizer + Wertausgleich. Mario Macke, Randersstr. 22, 2300 Kiel I.

Verk. Module: Moonsweeper 49,– DM, Parsec 39,– DM, Donkey Kong 49,– DM, Masch 29,– DM, TI-Invaders 19,– DM, 3 Literatur: Tips + Tricks 19,– DM, AG BIT MC-K.B. 39,– DM, + Zubehör: Rec.Kabel 19,– DM, Joy-Adapter 19,– DM. Angebote an: M.Vill, Birkhuhnweg 30, 8045 Ismaning.

---

32 KB-Speichererweiterung m. Centronicschnittstelle (externe Erw.) und Druckerkabel für 230,– DM zu verk. Peter Hielscher, Am Wall 22, 4401 Saerbeck.

TI-Bücher, jeweils 15,– DM: Tips & Tricks, Spielen + Lernen + Arbeiten, Farbe + Grafik + Ton + Spiele, Kochbuch I, Kochbuch II, Spielprogr. selbst erstellen I, Spielprogr. selbst erstellen II. Peter Hielscher, Am Wall 22, 4401 Saerbeck.

Verk. TI 99/4 A (32 K Byte) P-Box, Disk-Contr. (DOS80), D-Laufw. (40 Tr), 80 Zeich-K., Joyst., Est.Basic, Forth, Edit-Assm., div. Literatur ges. 1500,– DM (od. einzeln). Tel: 07191/22385 ab 18 Uhr.

Tips & Tricks für TI 99/4 A, Teile I + II, jeweils 13,5 KB (Ex-Basic). Progr. auf Kass. geg. Einsendung von 10,– DM. Peter Hielscher, Am Wall 22, 4401 Saerbeck

---

Lotto-Systeme für 6/49 und 7/38 mit Gewinngarantie. Ex-Basic-Programme auf Kass. geg. Einsendung von 10,– DM. Peter Hielscher, Am Wall 22, 4401 Saerbeck.

TI Box + Contr. + 32 K + Int. Dsk + Ext. Dsk + Schrank, evtl. 130 volle Disks, fast alle TI-Progr. VB. Tel: 02041/28 321.

TI kompl. Kons.-Box, 2 Dsk-Ex-Spr. Synty usw. alle Progr. für TI. ca. 120 in Assembler, 2700,– DM. Tel: 02041/28 321.

Achtung! Ganze Anlage von TI mit allen Progr. für TI. 130 Dsk mit ca. 400 Progr., 120 in E/A, DM 2700,–. Tel: 02041/28321.

TI + Tirec + Synty + Ex.–Base + Ex.Buch + TI-Joystick + ca. 20 Cass., Spiele + 50 Computerhefte + 4 Bücher, 650,– DM, Tel: 02041/28 321.

Verk. 32 KB intern, orig. TI 250,– DM. Suche o. tausche P.-Code. Karte + Pascal. Tel: 0209/390043 nach 14 Uhr.

Achtung! P.-Box + Floppy + 32 K + RS232 + Disk-Manager II, nur 1000,– DM. Mini-Mem + deutsch. Handb. 170,– DM, Ex-Basic 190,– DM, TI-Drucker 200,– DM, Konsole 220,– DM. 55 Spielmod. + 3 Cass. VB, 5 Bücher 100,– DM. Tel: 06121/844 659.

---

32 KB + Einbau in Konsole 210,– DM + Porto. Resettaste + Einbau in Konsole 25,– DM + Porto. Beides zusammen 220,– DM + Porto Tel: 02181/490396.

Verk. 32 K für TI. Tel: 068 06/490116.

TI 99/4 A + Ext-Basic + Handbuch (engl.) + Joysticks + Rec.-Kabel + Kass. + Literatur für nur 230,– DM. Tel: 0214/53388.

TI 99/4 A, Wer will mit mir Programme tauschen (TI u. Ex). Gratis Info. Einfach Postkarte an: M. Schenk, Rosenweg 4, 7107 Bad Friedrichshall 2.

Suche günstig für TI 99/4A RS232-Schnittstelle, 32 K RAM ext. Disk-Controller für BASF-Laufw. AG6138 o. 6106. J. Unruh, Th.-Fontane Str. 1, 2875 Ganderkesee I.

Verk. kopierte Listings: I,– DM o. 2 öS. Erwin Kinslechner, Eibesbrunnergasse 1/10, A-1120 Wien.

Verk. TI 99/4 A, TI-Rec. u. Kab., 4 Module (2 Spiele u. 2 Datenverw.). Progr. auf Kass. Alles orig. TI, Top Zustand 60,– DM. Günter Rybczynski Goethestr. 15 a, 5216 Niederkassel 2.

Verk. TI 99/4A + Peribox + BASF-Floppy + Ex-Basic + RS232 + Cass.-Kabel + 32 K 3 div. Module + Bücher, VB 3500,– DM. H. Ofner, Kreuzgasse 22, A-8580 Köflach.

---

32 K-Byte für TI-Peribox nur 199,– DM. 56 K Byte G-Ram-Karte nur 299,– DM. R. Moor. Tel: 02381/789030.

Verk. P-Box + Karte + Laufw. + Controller 700,– DM, Epson MX 80, neu 700,– DM, RS232 250,– DM, 32 K-Erw. 250,– DM, TI-Writer 250,– DM, Multiplan 250,–DM, Sp.-Edt. + Sp.-Synth. 110,– DM, Md.-Datenverarbeitung und vieles mehr. Tel: 0871/42178.

Biete Software aus Cass. in TI und X-Basic zu Superpreisen an: Infos bei: Tel: 04956/1402.

Suche folgende Steckmodule: Car Wars, Hopper, Parsec, Congo Bongo, Moon Patrol, Buck Roger, Pole Position, Burgertime. Tausche dafür 50 - 150 Topspiele aus meiner Spielesammlung. Infos bei: H. Duis, Lindenstr. 4, 2912 Uplengen I.

Databecker: Tips & Tricks 20,– DM, Spielen, Lernen, Arbeiten 15,– DM, Chip: TI Progr. 9,– DM, Alonpoole 77 Progr. 15,– DM. Tel: 06182/26186.

Verk. TI 99/4 A + P-Box + 32 K + Ext. Bas + Schachmodul + Apesoft + Grafik + Bücher (TI-Intern, Ed/Ass, TI-Special II) für 1700,– DM (auch einzeln). Tel: 02361/26460.

Das Kochbuch f.d. TI 10,– DM, Cass.Kabel 10,– DM, Basic für Anfänger (C) 5,– DM. Tel: 06638/1503 ab 15 Uhr.

EFFEKTIVER LISTSCHUTZ FÜR BASIC PROGRAMME

Da die Listschutzmethode, die Texas Instruments integriert hat, schon lange „überholt" ist, habe ich versucht die Schutzmethoden, die sich für Basic Programme eignen, zusammenzustellen.
Natürlich muß darauf hingewiesen werden, daß kein Listschutz perfekt ist und das Programm immer „unantastbar" bleibt!

I Der „alte" TI-Listschutz – und was man damit machen kann:

Wie ihr wahrscheinlich wissen werdet, befindet sich das Byte, das den Protected Zustand einleitet (genau: "BASIC Flag Byte") auf der Hexadezimalen RAM-Adresse 8345 (Dezimal: –31931).
Da aber das RAM-Scratch Pad (so heißt der Bereich von 8300 bis 83FF) nur teilweise dekodiert wurde, befindet sich das gleiche Byte auch auf 8245 (–32187).
Da diese Adresse nicht so bekannt ist wie 8345, kann man im Programm eine Abfrage mit CALL PEEK (–32187,A) realisieren und man erhält in A den Wert dieser Speicherstelle.
Dieser Wert sollte, bei einem geschützten Programm, bei 128 liegen!
Falls im Programm irgend ein Statement vorkommt, das

1. Fehlerbehandlung (ON WARNING NEXT/STOP)
2. Fehlersuche (TRACE/UNTRACE)
3. autom. Zeilennumerierung (NUM)
4. oder ON BREAK NEXT

ausführt, verändert sich das Basic Flag Byte dementsprechend. Dies kann man wie folgt errechnen: (vgl. Tabelle!) Ist BIT 0 gesetzt (d.h. Pgm ist Protected) hat die Adresse den Dezimalwert 128.
Kommt nun im Programm (vor der CALL PEEK Abfrage) ein ON BREAK next oder anderes der aufgeführten Statements, wird der Wert zu den 128 addiert!
Alle anderen Werte können aus der Tabelle entnommen werden.
In unserem Beispiel ergibt sich der Wert 192 (128+64). Hier aber nun die Tabelle der BIT Belegung der Adresse 8345:

BIT	0	1	2	3	4	5	6	7
DEZ	128	64	32	16	8	4	2	1
Basic Bed.	Protec	On Break Next	Un-Trace	Trace	?	On Warn. Stop	On Warn. Next	NUM

Jeder, der sich jetzt das Programm anschaut (– während der PRESCAN Abfrage CELAR (FCTN 4) drücken, bis sich der Computer im Command Mode befindet, und, falls man eine Speichererweiterung besitzt, führt man nun ein CALL LOAD (–31931,0) aus und schon kann man das Programm listen –) wird nicht viel Verdacht schöpfen, wenn er das CALL PEEK (–32187,A) entdeckt . . .
Dies eignet sich besonders gut für Programme, die viel mit PEEK & POKE arbeiten!
Wenn nun nochmals gestartet wird und der Wert vom Basic Flag Byte nicht mit dem Sollwert übereinstimmt, kann man den Computer abstürzen lassen.
Nun gibt es zwei Methoden, den TI ins Nirwana zu schicken:

a) Methode für 32 K RAM:

```
10 CALL CLEAR
20 CALL SCREEN(x)
30 CALL PEEK (–32187,A)
40
50          Statements
60
200 GOSUB 5555
5555 IF A<>ABS (–128)
*SGN(A) THEN CALL
LOAD (–31804,107,0)
5560 Return
```

b) Methode ohne 32 K RAM:

```
10 CALL CLEAR
20 CALL SCREEN (x)
30 CALL PEEK (–32187,a)
40 .
50 :        Statements
60 ∇
200 GOSUB 5555
5555 IF A <>ABS (–128)
*SGN(A) THEN RUN
"CS99"
5560 RETURN
```

Die Programme kontrollieren das Basic Flag Byte und springen (mit 32 K RAM) auf die Speicherstelle, die auf den User defined Interrupt (83c4) steht!
Ohne 32 K RAM läuft das ganze ohne LOAD ab – mit fast dem gleichen Ergebnis.
Hier springt das Programm, falls der Wert auf 8245 nicht 128 ist, auf die Kassettenroutine im GROM.
Diese Routine löscht zuerst das Programm (genauer: die Zeilenzeiger auf 8330_H und 8332_H!) und kontrolliert dann erst, ob die Parameter (CS 99') stimmen.
Und da ja keine 99 Rekorder angeschlossen werden können, springt er wieder ins BASIC zurück.
Dies zu der ersten Listschutzmethode. Wie ihr später sehen werdet, kann man diese Methode mit anderen verbinden . . .

II Befehlszähler als Listschutz?

Ein Befehlszähler ist nicht einer, der beim Bund die Befehle zählt, sondern eine Adresse (ebenfalls im RAM-Scratch-Pad), in der die Anzahl der eingegebenen Befehle (im Command Modus) gezählt werden!
Dieser Zähler lieg auf 8347_H (-31929_D). Nach einem NEW oder wenn das X-Basic angewählt wird, steht er auf 0.
Falls nun ein Diskettenlaufwerk angeschlossen ist und auf der Disk. sich ein Programm befindet mit dem File-Namen LOAD, wird dieses, ohne den Befehlszähler zu verändern, eingeladen. Dadurch ergibt sich die Möglichkeit, ein Programm direkt von der Diskette einzuladen und im Programm kann man prüfen, ob unterbrochen ist.
Die Abfrage könnte im Programm so realisiert werden:

```
10 CALL CLEAR
20 CALL SCREEN (x)
30 CALL PEEK (–31929,A)
40 :
50 :        Statements
60 ∇
200 GOSUB 5555.
5555 IF A<>0 THEN
CALL LOAD
(–31804,0,34)
5560 RETURN
```

In Zeile 5555 testet der Computer, ob die Variable A den Wert 0 hat, wenn dies nicht der Fall ist, springt der Computer auf das Titelbild!
Dieses Zurückspringen wird wieder über den Interrupt realisiert, der an die ROM Adresse >0022 springt!
Ohne Speichererweiterung kann man dies zwar

nicht so elegant lösen, aber im Endeffekt passiert das gleiche wie oben . . .
Anstatt CALL LOAD wird folgende Zeile geschrieben:
5555 ON BREAK NEXT :: IF A<>0 THEN GOTO 5555
Soweit zum Befehlszähler als Listschutzmethode.

III Der Programm (Zeilen-) Such-mich-Schutz:

„Versteck spielen mit dem Listing?", werdet ihr vielleicht jetzt sagen – und genauso ist es!
Das Programm listet nur seine erste Zeile – die anderen bleiben verborgen...
Man muß aber nicht unbedingt das ganze Programm verschwinden lassen, es reicht ja auch einige Zeilen – und der Betrachter schöpft nicht so leicht Verdacht . . .
Wie geht man dabei vor?
Um das ganze Programmlisting verschwinden zu lassen, kann man den Computer so manipulieren, daß er die erste Zeile mit der höchsten Zeilennummer versieht (=32767). Wenn er jetzt auf die nächste Zeile trifft, deren Nummer ja kleiner als 32767 ist, hört der TI auf zu listen!
Aber wie macht man denn das?
Um nun die erste Zeile aus einem Programm zu erhalten, geben wir folgendes ein:
CALL PEEK (–31950, A,B) – in A ist nun das Hi-Byte, d.h. die Zahl wird mit 256 multipliziert, in B befindet sich das Low-Byte der Zeilennummer.
Warum gerade –31950?
– Ganz einfach, auf Hexadezimaladressen 8330 und 8332 stehen die Zeiger für jeweils Beginn und Ende der Zeilenliste!
Nun wird aus A und B die erste Zeile errechnet und diese mit den Werten 127 und 255 in die letzte verwandelt.
(32767=127*256+255)
Nochmals alles zusammen:
1. CALL PEEK (–31950, A,B) – erste Zeile ermitteln
2. CALL LOAD (A*256 +B–65536–3,127,255) – auf die Speicherstelle, wo normalerweise die „richtige" Zeilennummer steht, 32767 schreiben!
Man muß aber erwähnen, daß diese Methode nicht gerade sicher ist, denn führt man ein RES aus, verschwindet der ganze Spuk wieder!
Etwas kann man dem jedoch entgegenhalten – indem man am Anfang des Programms ein ON BREAK NEST (oder: CALL LOAD (–31931, 64) und am Ende (z.B. Nach der Frage: "Noch ein Spiel?") ein CALL LOAD (–31804,0,34) ausführt. Natürlich gibt's noch mehr schöne Pokes, die den TI in die Weiten seines Speichers stürzen lassen . . .
Der Interupt steht hier gerne zur Verfügung!
Doch dazu kommen wir etwas später . . .
Um nur einen Teil eines Listings verschwinden zu lassen, gehen wir so vor:
Man schreibt den Programmteil, der später geschützt werden soll – und speichert diesen dann ab (Diskette!).
Nun schreibt man das restliche Programm und saved es mit dem MERGE Befehl ebenfalls auf Diskette ab.
Dies geschieht so:
SAVE DSK1.PGM1, MERGE. Das Diskettenlaufwerk springt an und speichert das Programm im DISPLAY/VARIABLE 163 Format!
So, nun liest man das Programm ein, das man schützen möchte. Es folgt die gleiche Prozedur wie oben!
1. CALL PEEK (–31950, A,B)
2. CALL LOAD (A*256 +B–65536–3,127,255)
Dann kann man listen und entdeckt nur die erste Zeile des ursprünglichen Listings . . . nun MERGEd man den anderen Programmteil dazu und listet nochmals!
Der Computer listet nun das dazu geMERGEde Programm! – Den anderen Teil unterschlägt er einfach . . . Der Betrachter schöpft so nicht so schnell Verdacht und ist ganz bestimmt etwas verdutzt, wenn er sich die Sprungadressen ansieht . . .

IV Der Zeilenchecksum-Schutz:

Wie die Überschrift aussagt, werden einzelne Zeilen und deren Inhalte abgefragt, ob der Sollwert noch besteht!
Dafür benötigt man eventuell wieder die Zeilenzeiger (8330 und 8332), um herauszufinden, wo das Basic-Programm im Speicher steht. Eines muß man noch erwähnen – für diese Schutzmethode ist eine Speichererweiterung Voraussetzung. Weshalb?
Ohne Ram-Zusatz werden die Programme ins VDP-RAM gelegt und ohne Erweiterung ist das VDP-RAM relativ tabu . . . (jedenfalls für XB-Programme).
Wie geht man vor?
Meistens liegen die Basic-Programme (wenn diese kurz sind) von –1500 bis –500 (Dezimalwerte) – es darf dann aber kein MC-Programm im Speicher stehen, sonst kann es vorkommen, daß das Basic-Programm auf A000 aufwärts steht! – Deshalb etwas auf die Zeilenzeiger achten.
Nun führt man eine FOR...NEXT-Schleife aus, um die Adressen und deren Inhalte zu erhalten.
Das ganze sollte dann etwa so aussehen:
FOR I=–1500 TO –500 :: CALL PEEK (I,A) :: PRINT I,A:" ";CHR$ (A) :: NEXT I
Es sollte natürlich auch ein kleines Programm im Speicher stehen, z.B. (zum Training reicht's. . .)

```
10   !Programmanfang
20   CALL CLEAR
30   CALL SCREEN(5)
40   !Programmende
```

Möchte man eine bestimmte Zeile im Programm abfragen, geht man in folgenden Schritten vor:
1. Mit Hilfe der Schleife die Zeile im Speicher suchen, die man abfragen möchte.
2. Hat man diese gefunden (auf die CHR$ am rechten Rand achten), notiert man die Adresse und die Werte der Inhalte.
Konkretes Beispiel:
– man möchte eine Rem-Zeile abfragen:
!(C) 1985 by OS-Soft
a) Schleife solange laufen lassen, bis man die Zeile entdeckt
b) dann eine markante Stelle merken (hier z.B. OS), die Adressen und die ASCII-Werte notieren (79/83)
3. An einer unscheinbaren Stelle im Programm wird die Abfrage wieder realisiert . . .
5555 CALL PEEK (Adr, A,B):: IF A+B<>162 THEN CALL LOAD (–31 952,255,231,255, 231)
Wenn man im Programm die Abfrage einbaut, muß man höchstwahrscheinlich die Adressen nochmals neu abfragen und abändern – diese verschieben sich etwas!
——BITTE DARAN DENKEN —— ZUERST SAVEN, DANN AUSPROBIEREN!!
Noch etwas unklar? – Das CALL LOAD? – die Werte, die auf die Zeilenzeiger hier gepoked werden –, bewirkt nichts anderes als ein NEW! Hat man sich zuvor die Werte gemerkt und poked die wieder auf 8330 und 8332, ist das Programm wieder da . . .
Da es manchmal so richtig schöne, interessante Abstürze gibt, hier einige zum Ausprobieren:
CALL LOAD (–31804, 0,34)
GPL RESET
(–31804,4,152)
zwar kein Absturz, aber eine Zeitverzögerung von 1 sec!
(–31804,2,40)

SERVICE

Bildschirm wird rot, fängt an zu flackern, . . . bei QUIT gibt es kein vollständiges RESET!!!
(–31804,10,50)
springt auf eine Sound-Ausgabe und hört sich auch dementsprechend an!
(–31862,0,0)
lädt das Programm von Disk mit dem Namen 'Load'!
Da ich diese Pokes auch einmal rausfinden mußte, wurde mein TI auch ganz schön strapaziert . . . aber bis jetzt „lebt" er noch . . .

V Manipulation mit Hilfe des Zeilenlistenzeigers:

Wie unter Punkt Drei wird auch hier der Zeilenlistenzeiger benötigt.
Für die Manipulation selbst braucht man die Speichererweiterung.
Wenn dann der Schutz aktiviert ist, kann dieser sowohl mit als auch ohne RAM-Erweiterung ausgeführt werden.
Weiterhin kann der Schutz mit anderen Methoden kombiniert werden ('Protected' und die Methode wie unter Punkt 3 beschrieben).
Was macht nun dieser Schutz? Wie wirkt er sich aus?
Nehmen wir an, daß wir ein BASIC Programm vor uns haben, das ganz „normal" (d.h. mit SAVE DSK1./CS1, PROTECTED) geschützt wurde.
Führt man nun ein CALL INIT :: CALL LOAD (–31931,0), bei angeschlossener Erweiterung aus, ist der ganze Spuk vorüber und man kann es ohne Probleme listen.
Hier setzt nun dieser neue Listschutz ein und läßt das Programm abstürzen. . . Wie macht man das nun?

```
a) CALL PEEK (-31952,
   A,B,C,D)
b) A=A*256+B-65536
   :: B=C*256+D-65536
c) FOR T=A TO B STEP 4
   :: CALL PEEK (T+2,
   C,D) :: CALL LOAD
   (C*256+D-65537,0)
   :: NEXT T
```

Hat man nun diese Befehlsfolge eingetippt, kann man das Programm saven (wer möchte, mit PROTECTED).
Ist dies geschehen, kann man sein Werk begutachten, indem man einfach LIST eintippt. Der Computer wird sich mit flakkerndem Bildschirm verabschieden.
Was verändert denn dieses kleine Programm?
Zuerst einmal wird mit CALL PEEK Abfrage unter a) die Lage der Zeilenliste im RAM ermittelt.
Teilen wir den Variablen zum Test einige Werte zu:
A=249 B= 34
C=249 D=193
So erhalten wir dann in b) folgende Werte für die Lage:
A=–1758 B=–1599
In diesem Bereich im RAM stehen also die Zeilenzeiger auf unser BASIC Programm.
Nun wird in Punkt c) die Lage der Zeilen ermittelt. Dafür muß man folgendes wissen:
Die Zeilenzeigerliste ist mit 4 Bytes aufgebaut:

0	10	253	194
0	20	253	204
0	30	xy	xyz

Zeilennummer RAM-Adresse

Um nun die Adresse einer Zeile zu erhalten, schaut man auf die Zeiger nach den Zeilennummern, die wieder per Highbyte und Lowbyte errechnet werden (Highbyte*256).
Dies wird in dem CALL LOAD Befehl gemacht.
Weiterhin wird mit diesem 'POKE' die Länge der Basiczeile auf Null gesetzt.
Wie das geschieht?
Normalerweise zeigt der Zeilenzeiger auf den ersten Befehl in einer BASIC Zeile. Hier wird aber nicht 65536, sondern 65537 subtrahiert und somit wird die im Grom vorhandene LIST Routine gestört, weil diese das Längenbyte unbedingt benötigt!
Somit kann man auch Zeilen kürzen, bzw. verlängern. *Oliver Siffrin*

BÖRSE

Verk. TI-Konsole 150,– DM, Ex-Basic 200,– DM, Handbuch deutsch 25,– DM, Pole Position 60,– DM, TI-Invaders 30,– DM, TI-Joyst. 35,– DM, 10 Progr.kass. 50,– DM, alle TI Revue 35,– DM, kompl. o. einzeln. Tel: 022 41/42978, 17 – 19 Uhr.

Achtung! Module! Achtung! Donkey-Kon9 65,– DM, Tomb-City 15,– DM, R.t. P. Isle 50,– DM, Burgertime 20,– DM, Q-Bert 40,– DM, MunchMan 25,– DM, Chisholm 15,– DM, Alpiner 20,– DM. Tel: 0208/23299 ab 18 Uhr.

Biete TI 99/4 A + 32 KB-Erw. + Turbo, Assembler + Joyst. Adapter + Kabel + Mon.Kabel + ca. 550 Progr. + 8 Module + Ext.-Basic, VB 820,– DM. Lief. per NN o. 3 Euroschecks. A. Schlaht, Große Str. 79, 2802 Ottersberg 1.

Verk. Cassetten: Benchmark-Test 10,– DM, 2 Spiele aus HC 1/84 10,– DM, Finanzberater Orig. TI 20,– DM, Marketing Spiel Orig. TI 20,– DM, Minimem-Progr. R. Majer 20,– DM, Pen-Shooting (f. Erw.) D. Karbach 10,– DM, Progr./Spiele TI Revue 2/84 10,– DM, Progr. und Spiele M. Foertsch 25,– DM, Oldies but Goodies 1 Orig. TI 20,– DM, Oldies but Goodies II Orig TI 20,– DM, Vokabel Trainer Dynamics 20,– DM, Progr./Spiele, Paedco Verlag 10,– DM, Video Archiv Dynamics 20,– DM. R. Thielmann, Tel: 02234/79646.

32 KRam Akku-gepuffert 210,– DM, Systemresettaste 20,– DM, Zeitlupe 60,– DM, Videoausgang 45,– DM, Zahlentastaturanschluss 40,– DM Netzteil +5/12V-SV alle 1A 94,– DM, 1-5 inkl. Einbau + Porto 325,– DM. Tel: 02181/490396 nach 19 Uhr.

TI 99/4 A. Suche Kontakt mit TI-Usern im Kreis Heilbronn (HN) wegen Clubgründung und Erfahrungsaustausch! M. Schenk, Rosenweg 4, 7107 Bad Friedrichshall, Tel: 07136/4116.

TI 99/4A. Suche CSP 99 (1 Laufw.) VB. Verk. RS232 (Extern) VB. Angebote an: Markus Schenk, Rosenweg 4, 7107 Bad Friedrichshall, Tel: 07136/4116.

Assembler Prg? Ich suche Tauschpartner! Material vorhanden. Melden bei: F. Gindullis, Drosselkamp 12, 2200 Elmshorn, Tel: 24243

HILFE! Wer kennt ZORK I und weiß, wie man zum Hades gelangen, das Boot (Pile of plastic) aufpumpen und den Dieb zuverlässig töten kann? Ralph Benzinger, Wilhelmstr. 65, 6800 Mannheim 51, Tel: 0621/794022.

Parsec 25,– DM, TI-Revue-Cass. 2 + 3/84 u. 7 + 8/85, je 5,– DM. TI-Special 2 (util. Ex) 9,–– DM, 3 Cass. voller Spiele 6,– DM. Tel: 06182/26186.

TI-Exbasic-Buch (engl.) 12,– DM, Computer Praxis 4184-2186 zus. 20,–DM. CK 8/9/84-10/85 zus. 20,– DM. TI-Special Nr. 4/86 7,– DM, Schachmodul 40,– DM. Tel: 06182/26186.

Um meinen TI 99/4A zu erweitern, verkaufe ich mein Phillips Telespiel mit 8 Module VB 295,--, T:05062/8089.

Suche diverse Hard- u. Software-Angebote (Listen etc.) an Michael Wank, Hauptstr. 60, 5204 Lohmar I.

Suche Pal-Modulator für TI, evtl. mit defekter Konsole. Gerald Sude, Tel.: 0711/379431 ab ca. 19 Uhr.

MONITOR-EIN UNGEWÖHNLICHES HILFSPROGRAMM

Ein ungewöhnlich leistungsfähiges Monitorprogramm, welches den TI-Debugger weit übertrifft, stellt das Programm „Monitor" von Jens Sundermann dar. Dieses Programm gibt einem die Möglichkeit, Maschinenprogramme auf jede Art zu kontrollieren und abzuändern, ohne den Monitor zu verlassen. Das wohl auffälligste und ungewöhnlichste an diesem Programm ist eindeutig der eingebaute Assembler. Er ermöglicht das Schreiben und Umändern von Maschinenprogrammen im Quellcode ohne den Umweg über den Editor/Assembler. So kann man eine Routine direkt mit den im Monitor integrierten Überwachungsmöglichkeiten direkt ausprobieren, verändern und verbessern. Um auch andere Programme auszuführen und zu überprüfen, ist im Monitor eine Routine eingebaut, welche es ermöglicht, vom Monitor aus Programme, die im Objekt-Code stehen, zu laden. Um diese Programme zu starten, hat man zwei Möglichkeiten: Den normalen Programmstart, wie er im Editor/Assembler unter „Run Program File" ausgeführt wird, oder den Single Step Betrieb.

EINGEBAUTE SPEICHER-ROUTINE

Um den normalen Programmstart durchzuführen, muß man zwei Daten über das auszuführende Programm wissen: Erstens die Startadresse, welche sich jedoch mit einem Befehl des Monitors beschaffen läßt, ohne sich eine Adresse merken zu müssen. Man muß nur den Startnamen kennen, da bei diesem alle sich im Speicher befindlichen Programmstartadressen mit dem zugehörigen Namen aufgelistet werden. Und zweitens, ob das Programm, das ausgeführt werden soll, im 32 oder im 40 Zeichen Modus des TI arbeitet, da man beide Modi vom Monitor aus ansprechen kann.

SOFTWARE IM TEST

Will man ein Programm im Single Step starten, so benötigt man ebenfalls die Startadresse des auszuführenden Programmes, wobei nun das Programm Befehl für Befehl abgearbeitet wird und so die Möglichkeit besteht, alle Register und Speicherstellen zu überprüfen, welche durch das Programm angesprochen werden. Ebenfalls wird der Befehl angezeigt, welcher gerade abgearbeitet wurde. Um ein in den Monitor geladenes, oder im Monitor geschriebenes, Programm aufzulisten, also zu disassemblieren, hat der Monitor eine eingebaute ;„Speicherroutine", womit alle Geräte, außer dem Kassettenrekorder, angesprochen werden können. Dadurch ist es auch möglich, ein Disassemblerlisting auf dem Drucker auszugeben, insofern man einen Drucker besitzt. Da dies der einzige Weg ist, den Drukker anzusteuern, kann man also nur die disassemblierten Befehle auflisten und nicht die dazugehörigen Speicherinhalte, welche dazu unumgänglich sind. Ebenfalls ist es unmöglich, Speicherinhalte allein auf dem Drucker auszugeben, was die auffälligste Unzulänglichkeitt des Programms bildet.

Der Monitor besitzt ebenfalls alle Befehle des TI-Debuggers, welche, da man für den Monitor TI-Konsole, 32 K, das Editor-Assembler Modul und mindestens ein Diskettenlaufwerk benötigt, bekannt sein dürften. Die Befehle sind fast alle identisch in der Arbeitsweise und Anwendung wie beim Debugger, jedoch ist die Bezeichnung dieser Befehle anders, da der Monitor eine größere Anzahl von Befehlen besitzt als der TI-Debugger. Nur ein Befehl wird wohl von einigen Anwendern vermißt: „CRU inspect/Change". Diesen muß man deshalb anhand eines kleinen Maschinenprogramms, was dank des eingebauten Assemblers möglich ist, simulieren.

SEHR BEDIENERFREUNDLICH

Der Monitor hat ebenfalls außer der Druckerausgabe noch einen kleinen, negativen Punkt. Er belegt im Low-Memory die Adressen 2676 bis 3F00 und im High-Memory die Adressen A000 bis A4AA. Man muß deshalb alle Programme, die man mit dem Monitor, oder bevor der Monitor geladen wird, laden und untersuchen will so abändern, daß eine Überschneidung der Adressen vermieden wird. Dies ist jedoch vor allem bei fremden Programmen, die man nur im Objekt-Code besitzt, problematisch, wenn nicht sogar unmöglich.

Sieht man jedoch davon ab, so bildet der Monitor von Jens Sundermann ein sehr bedienerfreundliches Programm, welches kaum einen Wunsch des Maschinenprogrammierers offen läßt und allen Anforderungen gewachsen ist, die man an ein Programm dieser Art stellt.

Oliver Huber

Die nächste TI-REVUE erscheint am 30. 5